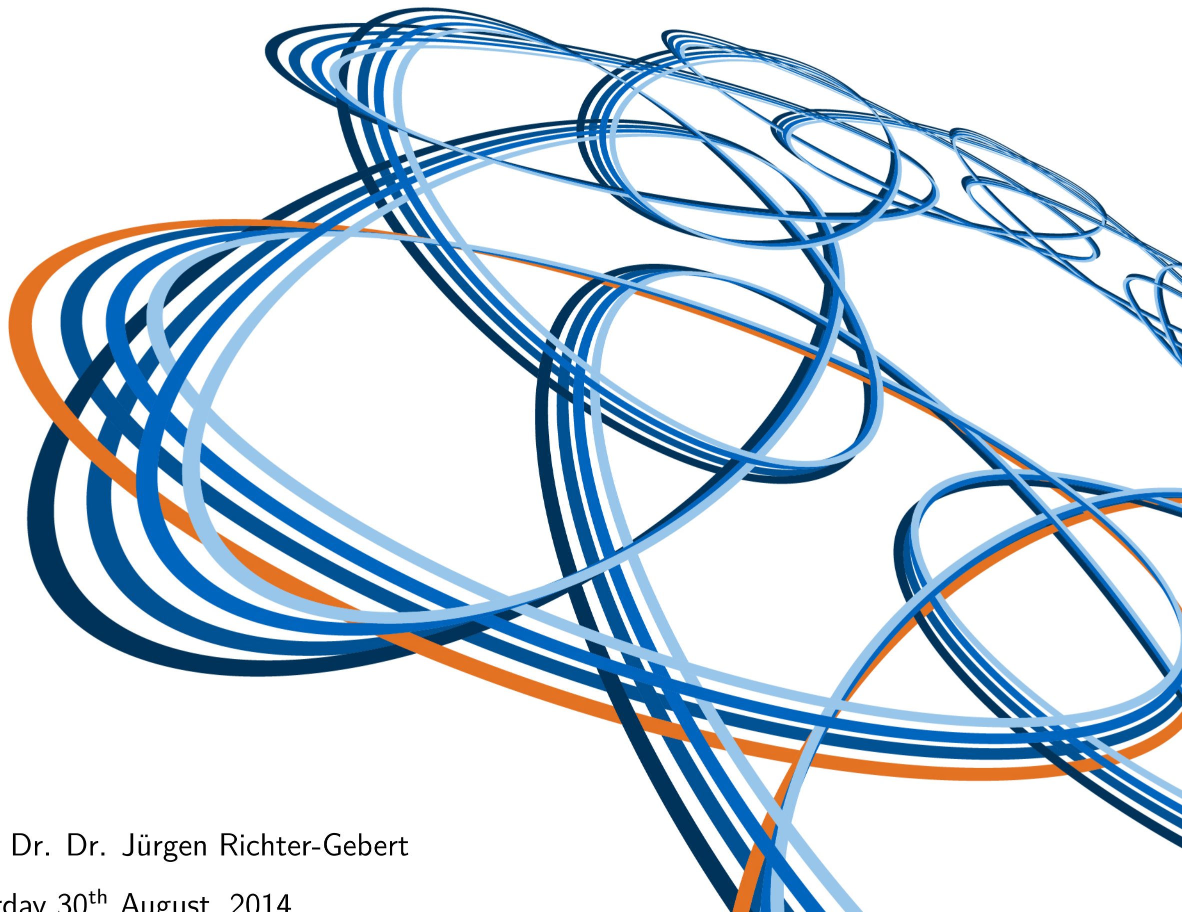
Technische Universität München

Fakultät für Mathematik
Lehrstuhl für Geometrie und Visualisierung

Interactive Image Sequences Converging to Fractals

Interaktive zu Fraktalen konvergierende Bildsequenzen

Bachelor's Thesis by Aaron Montag



Advisor & Supervisor:	Prof. Dr. Dr. Jürgen Richter-Gebert
Submission Date:	Saturday 30th August, 2014

I hereby confirm that this is my own work, and that I used only the cited sources and materials.

Guwahati, Assam, India

Abstract

We introduce a set of *rendering methods for fractals* by the *iterated deformation and composition of images.* These methods form the foundation for an effective and interactive GPU-based fractal visualization environment.
The first part discusses such methods for the fractals generated by *Iterated Function Systems* (IFS) and details a generalized escape time algorithm, that can be also used to visualize Julia sets. These algorithms iteratively apply a single operation on a texture in order to obtain the image of a fractal in convergence.
The second part is about *Kleinian groups* and their limit sets. A relation between the representation of group elements as string over group generators and the limit set will be derived. Utilizing this relation and assuming that suitable conditions hold, one can use a deterministic finite automaton that decides how to deform a small set of textures iteratively to obtain an image of the limit set.
Both parts end with the introduction of a measure based variation of these methods.

Zusammenfassung

In dieser Arbeit wird eine Klasse von Verfahren vorgestellt, welche durch die *iterierte Deformation und Überlagerung von Bildern* eine *Generierung von Fraktalen* ermöglicht. Diese Verfahren bieten Grundlage für effektive GPU-basierte interaktive Visualisierungsumgebungen für eine weite Klasse von Fraktalen.
Im ersten Teil der Arbeit werden Fraktale betrachtet, die durch *iterierte Funktionensysteme* (IFS) erzeugt werden. Es werden verschiedene Ansätze entwickelt, mit denen diese Fraktale durch die sukzessive Deformation und Überlagerung von Texturen erzeugt werden können. Ein verallgemeiner Fluchtzeit-Algorithmus, welcher auch zur Visualisierung von Julia-Mengen verwendet werden kann, wird in diesem Zusammenhang hergeleitet.
Im zweiten Teil wenden wir uns *Kleinschen Gruppen* und deren Limesmengen zu. Die Repräsentation der Gruppenelemente als Zeichenketten über die Erzeuger der Gruppe wird in Relation mit der Limesmenge gebracht. Diese Beziehung ermöglicht es unter geeigneten Bedingungen mithilfe eines deterministisch endlichen Automatens mehrere Texturen simultan zu deformieren, um schließlich in Konvergenz eine Darstellung der Limesmenge einer Kleinschen Gruppen zu erhalten.
In beiden Teilen wird abschließend eine auf Maßen basierende Abwandlung eingeführt.

Contents

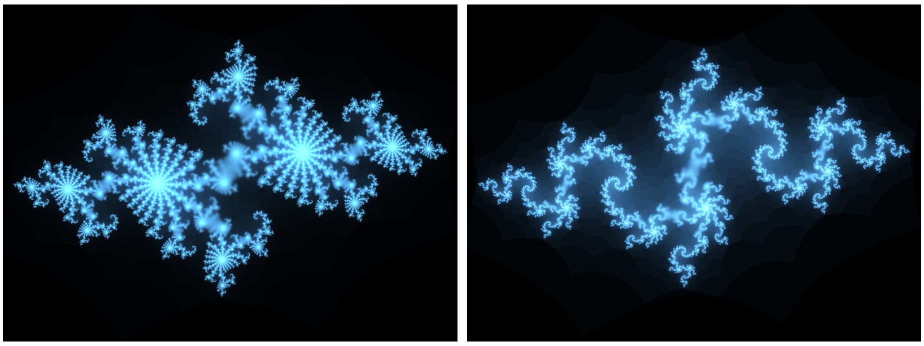


Figure 1: Screenshots of Felix Woitzel's demonstration.

1 Introduction

The seed for this thesis came from Felix Woitzel in 2011 when he published his WebGL Experiment *Progressive Julia Fractal* on the website *Chrome Experiments*: `http://aaron.montag.info/ba/1`.

Before continuing reading this thesis, the reader might check out Woitzel's experiment[1] in order to get the spirit of this beautiful underlying idea, which was planted into our minds last year.

So, what do we see in Woitzel's demonstration? We observe how the Julia-set is progressively built up by iteratively deforming and slightly lighting up a texture. The way the texture is deformed is controlled by the position of the cursor, and if we change some parameter, we can track how any change gradually propagates in the dynamic image seen.

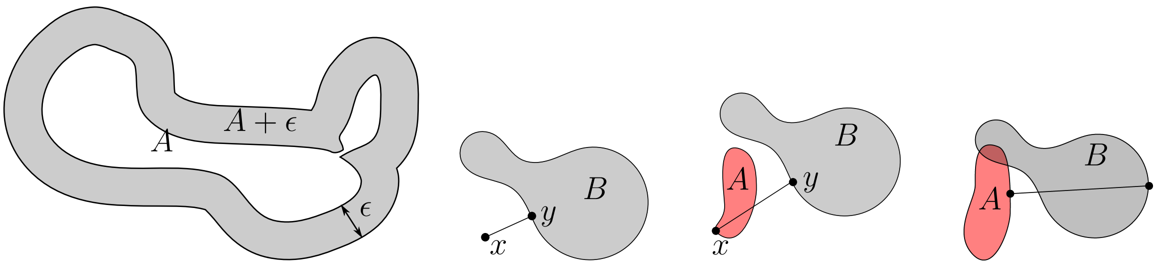
In August 2013 Professor Jürgen Richter-Gebert, who had discovered Woitzel's demonstration and already had developed some adaptions of it, told me about Woitzel's impressing idea and presented me with the opportunity to investigate this technique.

After playing around with this a little I got very enthusiastic and soon adapted the approach to render different fractals. The aim of this thesis is to explain Woitzel's approach and show several variations in order to render a wide class of fractals.

In Sections 2 and 3 we investigate a very general class of fractals, namely limit sets of *hyperbolic iterated function systems* (IFS), and modify Woitzel's idea of feedback loops in order to render those fractals. Then in Section 3.2, we examine *escape time algorithms* which give some better attempt in rendering limit sets of hyperbolic IFSs, and enable us to render a wider class of fractals, that includes Julia sets. In Section 3.3 a measure-theoretic approach using feedback loops that yields very good results for IFSs is given.The concept of limit sets of hyperbolic IFSs will be generalized in Section 3.4.

In Section 4 we will investigate Kleinian groups and their limit sets, which seems much harder at first glance. We will prove a convergence theorem which is probably new. A relative complex algorithm using this theorem is devolved. In the end of Section 4 we will enhance this algorithm by another measure-theoretic approach.

[1] WebGL for your browser is required in order to see Woitzel's demonstration and several other implementations in this thesis. For more information on installing WebGL visit `http://get.webgl.org`.



(a) $A + \epsilon$ (b) $d(x, B)$ (c) $d(A, B)$ (d) $h(A, B)$

Figure 2: Illustrations for various definitions

2 The Hausdorff Metric

We start by defining the metric space $(\mathcal{H}(X), h)$. It is the space that contains the fractals covered in this thesis.

Notation 2.1. Let (X, d) be a metric space, $A \subset X, \epsilon \in \mathbb{R}_{>0}$. Then

$$A + \epsilon := \{x \in X \mid \exists a \in A : d(x, a) \leq \epsilon\}$$

denotes the set A blown up by ϵ (see Figure 2a).

Definition 2.2 (The metric space $(\mathcal{H}(X), h)$)**.** Let (X, d) be a metric space. By

$$\mathcal{H}(X) := \{C \subset X \mid C \neq \varnothing \text{ compact}\}$$

we denote the set of all nonempty compact sets in X. $\mathcal{H}(X)$ can be equipped with a metric h as follows: Let $x \in X$ and $A, B \in \mathcal{H}(X)$. We set (for illustrations see Figures 2b to 2d)

$$d(x, B) := \inf_{y \in B} d(x, y) = \inf\{\epsilon \in \mathbb{R}_{>0} : x \in B + \epsilon\}$$

to be to the distance from x to B and

$$d(A, B) := \sup_{x \in X} d(x, B) = \sup_{x \in X} \inf_{y \in B} d(x, y) = \inf\{\epsilon \in \mathbb{R}_{>0} : A \subset B + \epsilon\}$$

to the distance from the set A to the set B, or in other words, the maximal distance a point of A has to be moved in order to lay in B. Note that $d(A, B) = 0$ is equivalent to $A \subset B$. Therefore $d(A, B) = d(B, A)$ does not hold in general, thus d does not form a metric on $\mathcal{H}(X)$. However, the *Hausdorff distance*

$$h(A, B) := \max\{d(A, B), d(B, A)\} = \inf\{\epsilon \in \mathbb{R}_{>0} : A \subset B + \epsilon \wedge B \subset A + \epsilon\}$$

defines a metric on $\mathcal{H}(X)$. It is straightforward to check that it satisfies the axioms of a metric. An explicit proof of the properties of a metric can be found in [Bar12]. If $h(A, B)$ is very small, then it is close to our intuition of A being almost the same set as B.
We say that a sequence $(A_n)_{n \in \mathbb{N}} \subset \mathcal{H}(X)$ of nonempty compact sets converges to the set $B \in \mathcal{H}(X)$ if $\lim_{n \to \infty} h(A_n, B) = 0$ and write $\lim_{n \to \infty} A_n = B$.

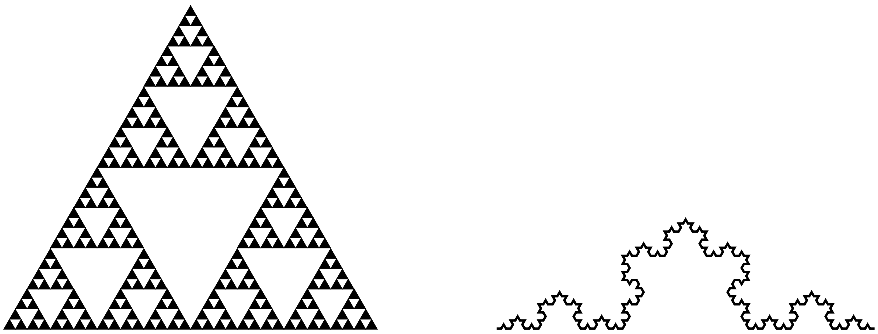


Figure 3: The Sierpinski triangle or the Koch curve consists of three or four copies of itself, respectively

3 Hyperbolic Iterated Function Systems

A striking property of fractals is their self-similarity. For instance, consider the two fractals in Figure 3, which both can be built of smaller copies of themselves.
At first glance, their explicit mathematical definition does not seem to be very plain. With the aid of hyperbolic iterated function systems it turns out that several fractals are uniquely determined by their self-similarities. From this approach arises a method to visualize those fractals interactively.

Definition 3.1 (hyperbolic iterated function system)**.** The triple $(X, d, \{w_1, \dots w_n\})$ is called a *hyperbolic iterated function system* (IFS) if the following conditions hold:

- (X, d) is a metric space.
- For every $i \in [n]$ [2] the function $w_i : X \to X$ is a contraction, that means there exists a Lipschitz constant $L_i < 1$ such that

$$\forall x, y \in X : x \neq y \Rightarrow d(w_i(x), w_i(y)) < L_i \cdot d(x, y)\,.$$

In his book [Bar12] Michael Barnsley gives a mathematical beautiful argument[3] to define the *limit set of an IFS*, which we will echo here. Later, those limit sets will become our fractals.

3.1 Convergence of the Iterated Hutchinson Operator

Definition 3.2 (Hutchinson operator)**.** Given a hyperbolic IFS $(X, d, \{w_1, \dots, w_n\})$. The so-called *Hutchinson operator* is defined as:

$$\begin{aligned} W : \mathcal{H}(X) &\to \mathcal{H}(X) \\ C &\mapsto \bigcup_{i=0}^{n} w_i(C) \end{aligned}$$

[2] In this thesis we will use $[n]$ to denote the set $\{1, \dots, n\}$

[3] John Hutchinson was the first to formalize the basic idea for this proof in [Hut79].

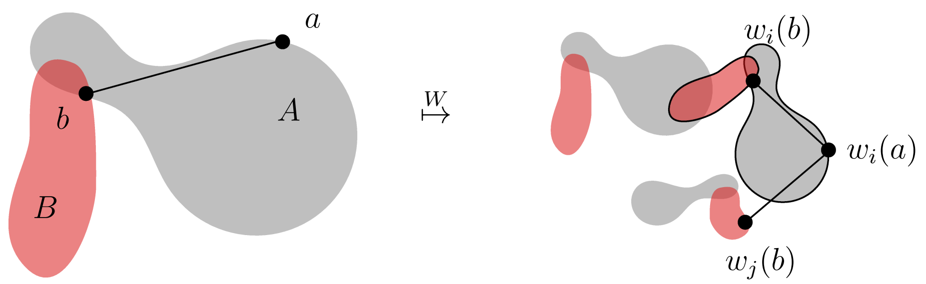


Figure 4: Visualization of Lemma 3.3

The Hutchinson operator is well defined, because (Lipschitz-)continuous functions map nonempty compact sets to nonempty compact sets (see [Kön03]) and the finite union of sets of this nature also is nonempty and compact.

Lemma 3.3. *Let $(X, d, \{w_1, \ldots, w_n\})$ be a hyperbolic IFS. Then the associated Hutchinson operator is a contraction on $(\mathcal{H}(X), h)$.*

Proof. Define $L := \max_{i\in[n]} L_i$, where L_i is the Lipschitz constant for w_i. As it is the maximum of a finite set of Lipschitz-constants less than 1, we also have $L < 1$. Let $A, B \in \mathcal{H}(X)$ and let us assume without loss of generality that $h(W(A), W(B)) = d(W(A), W(B))$ (Remember that we defined $h(A, B) := \max\{d(A, B), d(B, A)\}$).
Then one can estimate: (illustrated in Figure 4)

$$\begin{aligned} h(W(A), W(B)) = d(W(A), W(B)) &\leq \sup_{i\in[n],\, a\in A}\ \inf_{j\in[n],\, b\in B} d(w_i(a), w_j(b)) \\ &\leq \sup_{i\in[n],\, a\in A}\ \inf_{b\in B} d(w_i(a), w_i(b)) \leq \sup_{i\in[n], a\in A}\ \inf_{b\in B} L_i \cdot d(a, b) \\ &\leq L \cdot d(A, B) \leq L \cdot h(A, B) \end{aligned}$$

Therefore $W : \mathcal{H}(X) \to \mathcal{H}(X)$ is a contraction.

□

If $(\mathcal{H}(X), h)$ is a complete space, then we would be in the lucky situation to apply the Banach fixed-point theorem and could obtain for each hyperbolic IFS a unique nonempty compact set Λ such that $W(\Lambda) = \Lambda$.
Now there is a very handy theorem about the metric space $(\mathcal{H}(X), h)$, which gives us an answer to our problem: The completness of X is carried over to $\mathcal{H}(X)$.

Theorem 3.4. *Suppose (X, d) is a complete metric space. Then $(\mathcal{H}(X), h)$ is a complete metric space. Furthermore if $(A_n)_{n\in\mathbb{N}} \subset \mathcal{H}(X)$ is a Cauchy sequence then its limit set can be characterized as*

$$\lim_{n\to\infty} A_n = \{x \in X : \textit{there is a Cauchy sequence } x_n \in A_n \textit{ that converges to } x\ \} \in \mathcal{H}(X)\,.$$

Proof. See [Bar12, Chap 2., Thm 1.] □

With this property of $\mathcal{H}(X)$ and Lemma 3.3 the Banach fixed-point theorem (see [Mat13]) leads us to:

Corollary 3.5. *Let* (X, d) *be a complete metric space,* $(X, d, \{w_1, \ldots, w_n\})$ *a hyperbolic IFS. Then there exists a unique nonempty compact* limit set $\Lambda \in \mathcal{H}(X)$ *such that* $W(\Lambda) = \Lambda$ *for the Hutchinson operator* W*. Furthermore, for every* $C \in \mathcal{H}(X)$ *holds*

$$\lim_{n \to \infty} W^n(C) = \Lambda$$

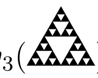
with respect to the Hausdorff distance h *on* $\mathcal{H}(X)$*.*

As from now we will always work on complete normed spaces.
So we can use this to characterize the famous Sierpinski triangle (see Figure 3).

Example 3.6 (The Sierpinski triangle)**.** Consider the subsequent hyperbolic IFS on $X = \mathbb{C}$ with the standard euclidean metric:

$$\begin{aligned} &w_i : \mathbb{C} \to \mathbb{C} \ \forall i \in \{1, 2, 3\} \\ &w_1 : z \mapsto \frac{1}{2} z \\ &w_2 : z \mapsto \frac{1}{2} z + 1 \\ &w_3 : z \mapsto \frac{1}{2} z + \frac{1}{2} + \sin(60°) \cdot i\,, \end{aligned}$$

where w_1, w_2, w_3 contracts points to 0, 1 and $\frac{1}{2} + \sin(60°) \cdot i$ respectively. Then the Sierpinski triangle with the vertices $0, 1, \frac{1}{2} + \sin(60°) \cdot i \in \mathbb{C}$ is a fixed point of W, because

$W($$) = w_1($$) \cup w_2($$) \cup w_3($$) = ($$) \cup ($$) \cup ($$) =$ .

Corollary 3.5 now attests us that the Sierpinski triangle is the unique fixed set for the hyperbolic IFS. And it allows us to start with an arbitrary nonempty compact set $C \in \mathcal{H}(X)$, from which we can obtain any arbitrarily good approximation of the Sierpinski by iterating the Hutchinson operator.

3.1.1 Texture Based Implementation of the Hutchinson Operator

From Corollary 3.5 immediately arises our first idea to render the limit set for a hyperbolic IFS $(X, d, \{w_1, \ldots, w_n\})$ on a two-dimensional space X.
We try to visually approach the limit set, for example the Sierpinski Triangle, as follows:

- Identify the two-dimensional space X with the pixels on the screen. Technically speaking, this is impossible, because pixels are discrete, whilst the space X might be continuous. So we will vaguely interpret some pixel as the point in X, which lies in the center of the quadratic region that is covered by this pixel.

- Sets in $\mathcal{H}(X)$ are encoded by images on the screen. We will paint a pixel white if its corresponding point in X belongs to the set, otherwise we will paint the pixel black.

- Our rendering process starts with an arbitrary frame, that shows a white shape on a black background, which corresponds to some set $C \in \mathcal{H}(X)$.

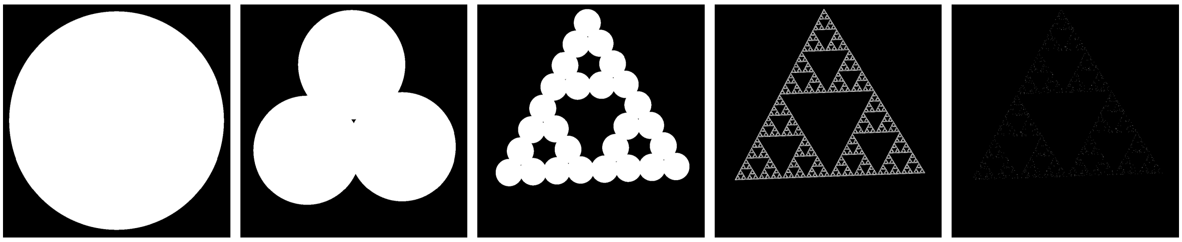


Figure 5: Screenshots of the set-based example implementation after 0, 1, 3, 9, 15 iterations respectively.

- Then we render new frames by deforming the last frame as the Hutchinson operator W would deform sets, i.e. we compute the overlay of the last frame transformed by w_i for $i \in [n]$. The new rendered frame is immediately shown to the user and meanwhile this process is repeated based on this currently rendered frame. This corresponds to iteratively applying W to C in order to compute $W^n(C)$.

  We deliberately stay vague in the technical details, such as how to treat points in the regions between the points in X which correspond to some pixel, because our discrete screen will never match the "reality" of sets on a continuous metric space. For instance, it might be a good guess to assume that a point of X belongs to the pictured set, if and only if the next point that is represented by a pixel belongs to the set.

- According to Corollary 3.5, after a while the user will see a picture which is arbitrarily close to the limit set with respect the Hausdorff metric.

- The user might change parameters of the IFS at running time, for instance, the rotation of the Sierpinski triangle. Regardless of which nonempty compact set was shown, we will, according to Corollary 3.5, again approach an image of the new limit set.

Some implementation of this approach to visualizing the Sierpinski triangle can be found here: `http://aaron.montag.info/ba/2`.[4]
After rendering a few frames, in this implementation something that is very close to the Sierpinski triangle becomes recognizable. But after another bunch of iterations, the set unfortunately vanishes. One explanation for this behavior is given by the fact that the Lebesgue measure of the Sierpinski triangle is zero, or to put it in other words, its Hausdorff dimension is strictly less than 2. As the number of white pixels closely correlates the Lebesgue measure of the pictured set, their number tends to 0. [5]
We need some better approach in order to visualize the limit set. In the next section we will develop a method that visually keeps track of all sets that occurred in the process of approaching the limit set.

[4] Press SPACE to set the current frame to some voluminous shape. You can stop the animation by clicking on the frame. Then by pressing CTRL a single frame gets rendered. The mouse position determines the rotation and scaling of the IFS.

[5] One can easily check that $\mathcal{L}(W^n C) \leq \left(\frac{3}{4}\right)^n \mathcal{L}(C)$ where $\mathcal{L}(\cdot)$ denotes the two-dimensional Lebesgue measure.

3.2 The Escape Time Algorithm

3.2.1 Introduction – Classical Approach for Filled Julia Sets

One of the most common and traditional applications of the escape time algorithms lies in rendering Julia or Mandelbrot sets.
We will shortly explain Julia sets. Given some function such as

$$\begin{aligned} J_c : \mathbb{C} &\to \mathbb{C} && \text{where } c \in \mathbb{C} \text{ is a fixed parameter} \\ z &\mapsto z^2 + c\,. \end{aligned}$$

Its *filled Julia set* is defined as

$$F_c := \{z \in \mathbb{C} \mid \lim_{n\to\infty} |J_c^n(z)| \neq \infty\}\,.$$

The filled Julia set is the set of those points $z \in \mathbb{C}$, for which, when iteratively applying the function J_c, all values remain bounded. Or, to utter it differently, we are looking for those points z in a dynamic system for which the sequence $(J_c^n(z))_{n\in\mathbb{N}}$ is not "attracted" to $\infty \in \hat{\mathbb{C}}$.
It can be observed that the augmentation of the absolute value

$$|J_c(z)| - |z| = |z^2 + c| - |z| \geq |z|^2 - |z| - |c| =: p(|z|)$$

can be bounded from below by a quadratic polynomial p, which increases monotonously for $|z| \geq R$, where $R = \frac{1}{2} + \sqrt{\frac{1}{4} + |c|}$ is its major root. Hence we can interpret $R \in \mathbb{R}$ as a bailout radius, i.e. points $z \in \mathbb{C}$ whose absolute value is greater than this radius R are mapped by J_c to points with even bigger absolute value, and even more, $\lim_{n\to\infty} |J_c^n(z)| = \infty$.
So, the standard approach to disprove that a given point $z \in \mathbb{C}$ belongs to the filled Julia set is to just iterate J_c on z until $|J_c^n(z)| > R$. The needed number of iterations until we land outside of this bailout radius is called the *escape time*. On the other hand, if $J_c^n(z)$ remains within the bailout radius for a large fixed number of iterations, e.g. 200, then it is likely that $z \in F_c$.
In order to visualize the filled Julia set, it is rather common to output those escape times for every point, instead of only showing the plain filled Julia set. We will generalize this concept such that we can also apply it to hyperbolic IFSs.

3.2.2 Generalization of the Escape Time Algorithm

Definition 3.7 (point determined function, preservation of inclusion, positive invariance)**.**
A function $f : \mathcal{P}(X) \to \mathcal{P}(X)$ is called *point determined* if[6]

$$f(A) = \bigcup_{a\in A} f(\{a\})\,.$$

However, we do not require $f(\{a\})$ to be a singleton. It also might be a collection of several points, or it might be the empty set. Note that the point determined nature implies *preservation of inclusion*, i.e. for every $A, B \in \mathcal{P}(X)$ with $A \subset B$ we get $f(A) \subset f(B)$.
We will call a set $A \subset X$ *positively invariant* under f if

$$f(A) \subset A\,.$$

[6]The power set $\mathcal{P}(X)$ is defined as the set of all subsets of X.

Example 3.8. The inverse Julia-map

$$\begin{aligned} J_c^{-1} : \mathcal{P}(\mathbb{C}) &\to \mathcal{P}(\mathbb{C}) \\ A &\mapsto \{z \in \mathbb{C} \mid J_c(z) \in A\} \end{aligned}$$

is obviously point determined. Furthermore $J_c^{-1}(\{z\}) = \{\pm\sqrt{z-c}\}$.
The closed disk $\overline{D}_R(0) = \{z \in \mathbb{C} \mid |z| \leq R\}$ is positively invariant under J_c^{-1} where $R > \frac{1}{2} + \sqrt{\frac{1}{4} + |c|}$ is some valid bailout radius: To see this take the contrapositive of the bailout radius property $z \notin \overline{D}_R(0) \Rightarrow J_c(z) \notin \overline{D}_R(0)$ which is equivalent to $z \in \overline{D}_R(0) \Rightarrow J_c^{-1}z \in \overline{D}_R(0)$.

We introduced the concept of positively invariant sets especially for the important case of hyperbolic IFSs:

Example 3.9 (Hutchinson operator for a hyperbolic IFS)**.** Let $(X, d, \{w_1, \ldots, w_n\})$ be a hyperbolic IFS. Clearly, the Hutchinson operator $W : \mathcal{H}(X) \to \mathcal{H}(X)$, $C \mapsto \bigcup_{i=0}^n w_i(C)$ is point determined. Furthermore, let L be the maximal occurring Lipschitz constant of the transformations w_i (as defined in Lemma 3.3).
If we knew the limit set Λ of the IFS, then we could choose $\Lambda + \epsilon$ for any $\epsilon \in \mathbb{R}_{>0}$ as a positively invariant set, because by Lemma 3.3 $h(W(\Lambda+\epsilon), \Lambda) \leq L \cdot d(\Lambda + \epsilon, \Lambda) = L \cdot \epsilon$, thus $W(\Lambda + \epsilon) \subset \Lambda + L \cdot \epsilon \subset \Lambda + \epsilon$.
There is another positively invariant set, that can be directly computed from the unique fixed points $Fix_1, \ldots Fix_n$ of $w_1, \ldots w_n$ respectively. Their existence and uniqueness is given by Banach's contraction mapping theorem. Let $0 \in X$ and $M = \max_{i \in [n]} d(0, Fix_i)$ its maximal distance to the fixed points, then the disk $\overline{D}_R(0)$ is positively invariant under W if $R \geq \frac{M(1+L)}{1-L}$. Just check that for any $z \in \overline{D}_R(0)$, $i \in [n]$:

$$\begin{aligned} d(0, w_i z) &\leq d(0, w_i Fix_i) + d(w_i Fix_i, w_i z) \leq d(0, Fix_i) + L \cdot d(Fix_i, z) \\ &\leq d(0, Fix_i) + L \cdot d(Fix_i, 0) + L \cdot d(0, z) \leq M(1+L) + L \cdot R \leq R\,, \end{aligned}$$

thus $w_i z \in \overline{D}_R(0)$. As $z \in \overline{D}_R(0)$ and $i \in \mathbb{N}$ was arbitrary, we have shown that

$$W\left(\overline{D}_R(0)\right) \subset \overline{D}_R(0)\,.$$

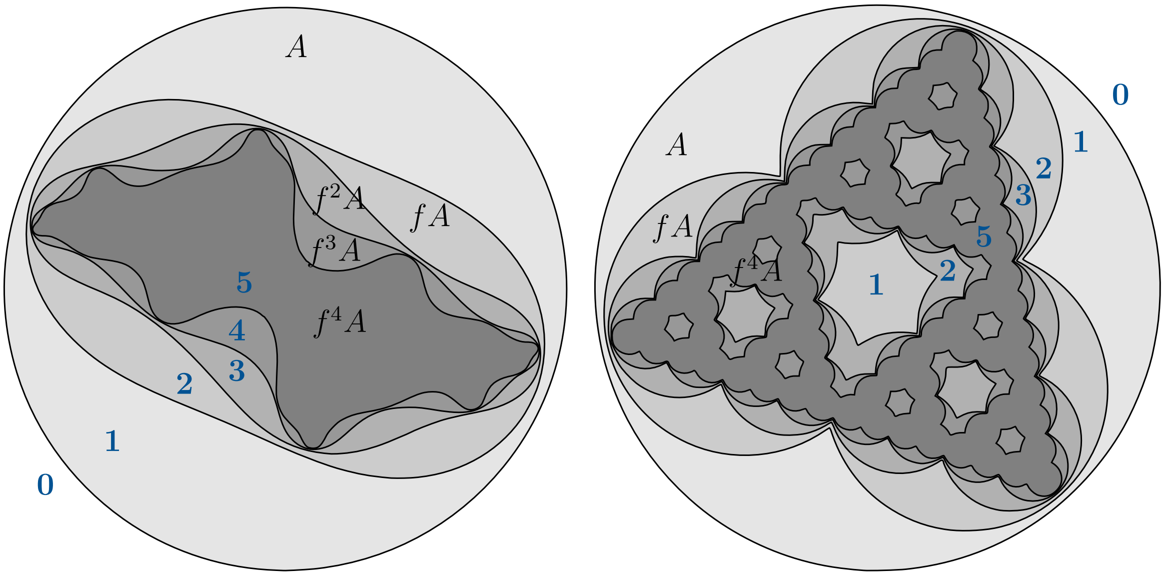
Lemma 3.10. *Let $f : \mathcal{P}(X) \to \mathcal{P}(X)$ be point determined and $A \in \mathcal{P}(X)$ positively invariant.*
Then f iterated on A generates an infinite decreasing sequence of sets in $\mathcal{P}(X)$:

$$A \supset f(A) \supset f^2(A) \supset f^3(A) \supset \ldots$$

and the set

$$\Lambda := \bigcap_{n \in \mathbb{N}} f^n(A)$$

is a fixed point of f, namely $f(\Lambda) = \Lambda$.
Furthermore, if $n \in \mathbb{N} : f^n(A) \in \mathcal{H}(X)$ for all $n \in \mathbb{N}$, then also $\Lambda \in \mathcal{H}(X)$ and the sequence of sets $f^n(A)$ converges to Λ with respect to the Hausdorff metric.



(a) The Julia set with $f = J_c^{-1}$ (b) The Sierpinski Triangle with $f = W$

Figure 6: The sets $A, fA, \ldots f^4A$, where A is some positively invariant disk. The corresponding values of `time` (see Definition 3.12) are labeled in **blue**.

Proof. **The decreasing subset-sequence property (see Figure 6):** By induction we show that $f^{n+1}A \subset f^nA$ for all $n \in \mathbb{N}$. The base case for $n = 0$, namely $fA \subset A$ holds due to our assumption that A is positively invariant under f. For the inductive step let us assume that $f^nA \subset f^{n-1}A$. Then $f^{n+1}A \subset f^nA$ follows immediately from the preservation of inclusion for point determined functions.

f has Λ as a fixed point: A simple calculation yields

$$f(\Lambda) = f(\bigcap_{n\in\mathbb{N}} f^n(A)) = \bigcap_{n\in\mathbb{N}} f^{n+1}(A) = \bigcap_{n\in\mathbb{N}} f^n(A) = \Lambda\,.$$

The second last equality holds since $f^n(A) \subset A$ for all $n \in N$.

If $f^n(A) \in \mathcal{H}(X)$, then $\Lambda \in \mathcal{H}(X)$: Now, let us assert that $\forall n \in \mathbb{N} : f^n(A) \in \mathcal{H}(X)$. Then Λ is closed as an intersection of closed sets.

Λ is bounded as a subset of the compact set A.

Every collection of closed subsets of the compact set A satisfies the finite intersection property, i.e: If the intersection over a finite number of closed sets in the collection is nonempty, then the collection has a nonempty intersection itself (see [Bro13]). For the decreasing sequence f^nA of nonempty compact sets this implies that $\Lambda := \bigcap_{n\in\mathbb{N}} f^n(A)$ is nonempty.

Hausdorff convergence of f^nA to Λ for $n \to \infty$: Note that $h(f^nA, \Lambda) = d(f^nA, \Lambda)$ since $\Lambda \subset f^nA$, or equivalently, $d(\Lambda, f^nA) = 0$. Now we will prove that

$$\lim_{n\to\infty} d(f^nA, \Lambda) = \lim_{n\to\infty} \inf\{\epsilon \in \mathbb{R}_{>0} \mid f^nA \subset \Lambda + \epsilon\} = 0\,.$$

Let us assume for contradiction that this convergence does not hold. Then there is an $\epsilon \in \mathbb{R}_{>0}$ and a sequence $x_n \in f^n A$ such that $d(x_n, \Lambda) \geq \epsilon$ for all $n \in \mathbb{N}$. As $(x_n)_{n\in\mathbb{N}} \subset A \in \mathcal{H}(X)$ there is a subsequence $(n_k)_{k\in\mathbb{N}}$ such that the subsequence $(x_{n_k})_{k\in\mathbb{N}}$ converges. Namely, set $x := \lim_{k\to\infty} x_{n_k} \in A$. The decreasing subset-sequence property yields that $x_n \in f^m A \,\forall m \geq n$, which, in combination with the closed nature of the sets $f^m A$, implies that $x = \lim_{k\to\infty} x_{n_k} \in f^m(A)$ for all $m \in \mathbb{N}$, therefore $x \in \Lambda$. But on the other hand, from the continuity of $d(\cdot, \Lambda)$ follows that $d(x, \Lambda) = \lim_{k\to\infty} d(x_{n_k}, \Lambda) \geq \epsilon$, which was a contradiction to $x \in \Lambda$.

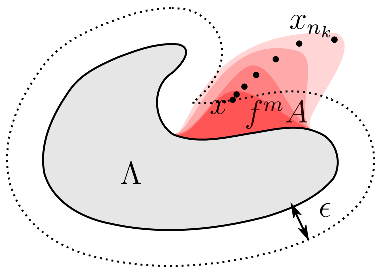


Figure 7: Illustration of the contradiction

□

Example 3.11 (Filled Julia sets)**.** Let us come back to Example 3.8. As it has been discussed, J_c^{-1} is point determined and $\overline{D}_R(0)$ is positively invariant, where $R \in \mathbb{R}$ is a valid bailout radius.
Since J_c is continuous, the preimages of closed sets are closed. Also the preimages of bounded sets are bounded. By the fundamental theorem of algebra these preimages of nonempty sets remain nonempty, thus finally $J_c^{-1}(C) \in \mathcal{H}(\mathbb{C})$ for all $C \in \mathcal{H}(\mathbb{C})$.
Now Lemma 3.10 states that the sequence $(J_c^{-n}\overline{D}_R(0))_{n\in\mathbb{N}}$ converges with respect the Hausdorff-Metric to the set

$$\Lambda = \bigcap_{n\in\mathbb{N}} J_c^{-n}\overline{D}_R(0) = \{z \in \mathbb{C} | \forall n \in \mathbb{N} : J_c^n(z) \in \overline{D}_R(0)\} = F_c\,.$$

What is about hyperbolic IFSs? The convergence in Lemma 3.10 does not give us anything new. But we might use the fact that the generated sets of the sequence approaching the unique limit set are contained in each other.
Now we are able to define a more generalized escape time, which also works for hyperbolic IFSs.

Definition 3.12. Let $f : \mathcal{P}(X) \to \mathcal{P}(X)$ be point determined and $A \in \mathcal{P}(X)$ be positively invariant.
Then we define the (discrete) escape time of this system (f, A) as

$$\begin{aligned} \texttt{time} : X &\to \mathbb{N}_\infty := \mathbb{N} \cup \{\infty\} \\ x &\mapsto \min\{n \in \mathbb{N} \mid x \notin f^n(A)\} \end{aligned}$$

where we use the convention $\min \varnothing = \infty$. $\texttt{time}(x)$ counts the number of sets among $A, fA, f^2A, \ldots$ that cover x. (For illustrations see Figures 6a and 6b).

The following lemma gives us a very handy recursive definition of $\texttt{time}(x)$:

Lemma 3.13. `time` *for the system* (f, A) *fulfills the following recursive description:*

$$\mathtt{time}(x) = \begin{cases} 0 & \textit{if } x \notin A \\ 1 & \textit{if } x \in A \wedge x \notin f(A) \\ \max\{\mathtt{time}(y) + 1 \in \mathbb{N}_\infty \mid y \in f^{-1}(\{x\})\} & \textit{otherwise} \end{cases} \tag{1}$$

Note that in our Julia set example 3.8 the expression $y \in f^{-1}(\{x\})$ for $f = J_c^{-1}$ is equivalent to the much more simple expression $y = J_c(x)$ and $x \notin f(A)$ becomes $J_c(x) \notin A$. Thus the formula above gives us the standard approach to calculate the escape time.
For a hyperbolic IFS $(X, d, \{w_1, \ldots, w_n\})$ the recursive expression for `time` with $f = W = (C \mapsto \bigcup_{i=0}^n w_i(C))$ might be harder to compute if the cardinality of the set $f^{-1}(\{x\}) = \bigcup_{i=1}^n w_i^{-1}\{x\}$ becomes infinite. But for several IFSs, for instance for the Sierpinski triangle (see Example 3.6), each of the w_is are injective and their inverse is easy to compute. Then evaluating the maximum in (1) can be done in $\mathcal{O}(n)$ steps.

Proof of Lemma 3.13. We will verify Equation (1) for every $x \in X$.

1st case: $\mathtt{time}(x) \in \{0, 1\}$ Equation (1) obviously holds by definition of `time`.

2nd case: $\mathtt{time}(x) \in \mathbb{N}_{\geq 2}$ So let $\mathtt{time}(x) = n \geq 2$. By definition of `time` and the decreasing subset-sequence property follows that

$$\forall k < n \quad x \in f^k(A) \quad \text{and} \tag{2}$$
$$\forall k \geq n \quad x \notin f^k(A) \tag{3}$$

In particular, $x \in f^{n-1}A$, which also has a preimage $y \in f^{n-2}A$ such that $y \in f^{-1}(\{x\})$. Then by definition $\mathtt{time}(y) \geq n - 1$. So we have established

$$\max\{\mathtt{time}(y) + 1 \in \mathbb{N}_\infty \mid y \in f^{-1}(\{x\})\} \geq n\,.$$

Let $y \in f^{-1}(\{x\})$. We will show by contradiction that $\mathtt{time}(y) < n$. Suppose that $k := \mathtt{time}(y) \geq n$. Then by definition $y \in f^{k-1}A$. We conclude with $x \in f(\{y\})$ that $x \in f^k A$, which is a contradiction to (3) since $k \geq n$. So we have proven the other inequality

$$\max\{\mathtt{time}(y) + 1 \in \mathbb{N}_\infty \mid y \in f^{-1}(\{x\})\} \leq n\,.$$

3rd case: $\mathtt{time}(x) = \infty$ Note that now we have

$$x \in \Lambda := \bigcap_{n \in \mathbb{N}} f^n(A)$$

and $f(\Lambda) = \Lambda$ by Lemma 3.10. So there is a $y \in \Lambda$ such that $x \in f(\{y\})$, or equivalently $y \in f^{-1}(\{x\})$. From $y \in \Lambda$ follows that $\mathtt{time}(y) = \infty$. Hence

$$\max\{\mathtt{time}(y) + 1 \in \mathbb{N}_\infty \mid y \in f^{-1}(\{x\})\} \geq \infty + 1 = \infty = \mathtt{time}(x)\,,$$

therefore (1) holds.

□

This recursion relation gives us the foundation for our first algorithm.
Suppose we are given some arbitrary function $\texttt{time}_0 : X \to \mathbb{N}_\infty$. Then one can define the sequence $(\texttt{time}_t)_{t\in\mathbb{N}}$ of functions from X to $\mathbb{N}_\infty$ as follows:

$$\texttt{time}_{t+1}(x) = \begin{cases} 0 & \text{if } x \notin A \\ 1 & \text{if } x \in A \wedge x \notin f(A) \\ \max\{\texttt{time}_t(y) + 1 \in \mathbb{N}_\infty \mid y \in f^{-1}(\{x\})\} & \text{otherwise} \end{cases} \tag{4}$$

Note that this equation is the same as (1) in Lemma 3.13, but we use $\texttt{time}_t$ in order to compute $\texttt{time}_{t+1}$.
By induction on t it turns out that

Corollary 3.14. *For $t \geq 1$ and $x \in X$ holds*

$$\texttt{time}_t(x) = \texttt{time}(x)$$

provided that $\texttt{time}(x) \leq t$ or $\texttt{time}_t(x) \leq t$.

Proof by induction on t. By definition and the fact that $\texttt{time}_0(x)$ has no negative numbers in its domain one has $\texttt{time}_t(x) = 0$ iff $\texttt{time}(x) = 0$ and $\texttt{time}_t(x) = 1$ iff $\texttt{time}(x) = 1$ for every $t \geq 1$. This also proves the base case for $t = 1$.
Assume for the inductive step $t \geq 2$. Let $\texttt{time}_t(x) \leq t$ and different from 0 or 1. Then $\texttt{time}_t(x) = \max\{\texttt{time}_{t-1}(y) + 1 \in \mathbb{N}_\infty \mid y \in f^{-1}(\{x\})\} \leq t$. It follows from the inductive assumption that $\texttt{time}_{t-1}(y) = \texttt{time}(y)$ for all $y \in f^{-1}(\{x\})$. Hence by Lemma 3.13 $\texttt{time}_t(x) = \texttt{time}(x)$. In the same manner one can show that $\texttt{time}(x) \leq t$ implies $\texttt{time}_t(x) = \texttt{time}(x)$. □

This corollary can be used for the subsequent GPU-based algorithm.

3.2.3 Texture Based Implementation of the Escape Time Algorithm

The following algorithm visualizes the elements of the sequence $(\mathtt{time}_t)_{t\in\mathbb{N}}$ as defined in (4). To omit technical details, we will consider a single channel texture as a map, which assigns each pixel some real number.

Algorithm 1: A method to calculate and visualize `time` progressively

```
Initialize two high resolution single channel textures CurrentTime and PreviousTime
  of the same size.
while program is running do at most (roughly) 30 times a second
    /* The rendering procedure for a single frame */
    Get user input, such as the current mouse coordinate and specified parameters.
    Based on this data calculate f and a corresponding positively invariant set A,
      ensuring that A lies entirely in the region covered by the textures.
    foreach pixel p_x on the texture CurrentTime do [7]
        x ← point in X that corresponds the pixel p_x
        if x ∈ A then
            time ← 1
            foreach y ∈ f^{-1}({x}) do
                if y ∈ A then
                    /* Here some interpolation might be used. */
                    p_y ← pixel(coordinate) that corresponds the point y
                    newtime ← PreviousTime(p_y) + 1
                else
                    newtime ← 1
                end
                time ← max(time, newtime)
            end
        else
            time ← 0
        end
        CurrentTime(p_x) ← time
    end
    Display the texture CurrentTime on the screen by, for instance, interpreting high
      values as bright colors.
    PreviousTime ← CurrentTime
end
```

If the user updates a parameter, he can instantly observe how his modification gradually propagates in the rendered image, while the structures that were rendered previously remain visible. Corollary 3.14 guarantees us that after $t \in \mathbb{N}$ rendered frames all pixels indicating an escape time which is less then or equal to t show correct escape time.

Implementation for the Filled Julia Set You can find some example implementation for filled Julia sets at `http://aaron.montag.info/ba/3`.

[7] The code in this loop usually is processed in parallel by the pixel shader (GPU)

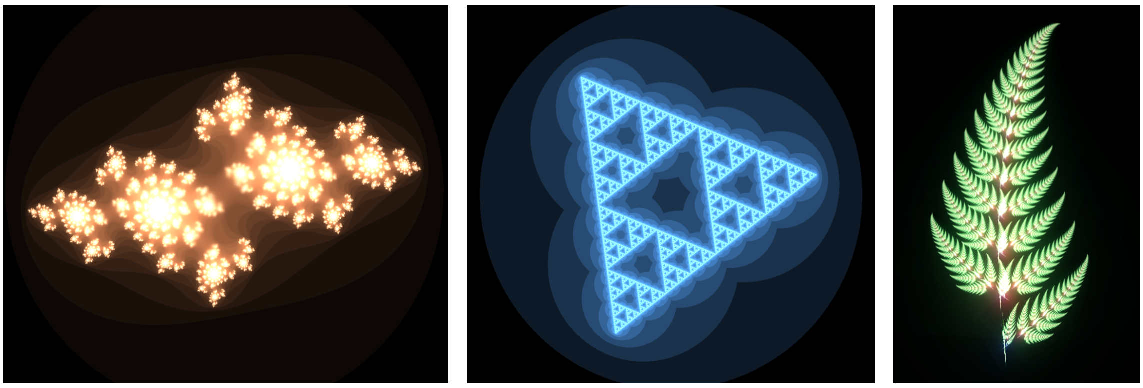


Figure 8: Screenshots of the implementation for the filled Julia set, Sierpinski triangle and Barnsley's farn respectively.

Here the user is able to set the parameter $c \in \mathbb{C}$ interactively by the cursors position. Note that the for-loop over the $y \in f^{-1}(\{x\})$ in line 9 can be replaced by $y = J_c(x)$.

Basically this implementation copies Felix Woitzels idea, despite the fact that he uses the entire screen as a positive invariant set instead of some disk representing the bailout radius.

Implementation for Hyperbolic Iterated Function Systems An implementation for the Sierpinski Triangle can be found here: `http://aaron.montag.info/ba/4`.

The position of the cursor again will be interpreted as some complex number $c \in \mathbb{C}$. From this we build the IFS on $X = \mathbb{C}$:

$$\begin{aligned}
&w_i : \mathbb{C} \to \mathbb{C} \ \forall i \in \{1, 2, 3\} \\
&w_1 : z \mapsto \frac{1}{2} z + c \\
&w_2 : z \mapsto \frac{1}{2} z + e^{i\frac{2}{3}\pi} c \\
&w_3 : z \mapsto \frac{1}{2} z + e^{i\frac{4}{3}\pi} c
\end{aligned}$$

So $y \in f^{-1}(\{x\})$ in line 9 becomes $y \in \{2x - c, 2x - e^{i\frac{2}{3}\pi} c, 2x - e^{i\frac{4}{3}\pi} c\}$.

Another hyperbolic IFS generates Barnsley's farn, which is built of four contracting affine transformations of itself: `http://aaron.montag.info/ba/5`. If you want to learn more about this fractal, see [Bar12].

Fractals Generated by Circle Inversions A bunch of interesting fractals can be built by a set of circle inversions. (For more details on circle inversions in general see, for instance, [Nee11].)

Let $C_1, \ldots, C_n$ be a set of circles in the plane with radii $r_1, \ldots, r_n$ and centers $c_1, \ldots, c_n$,

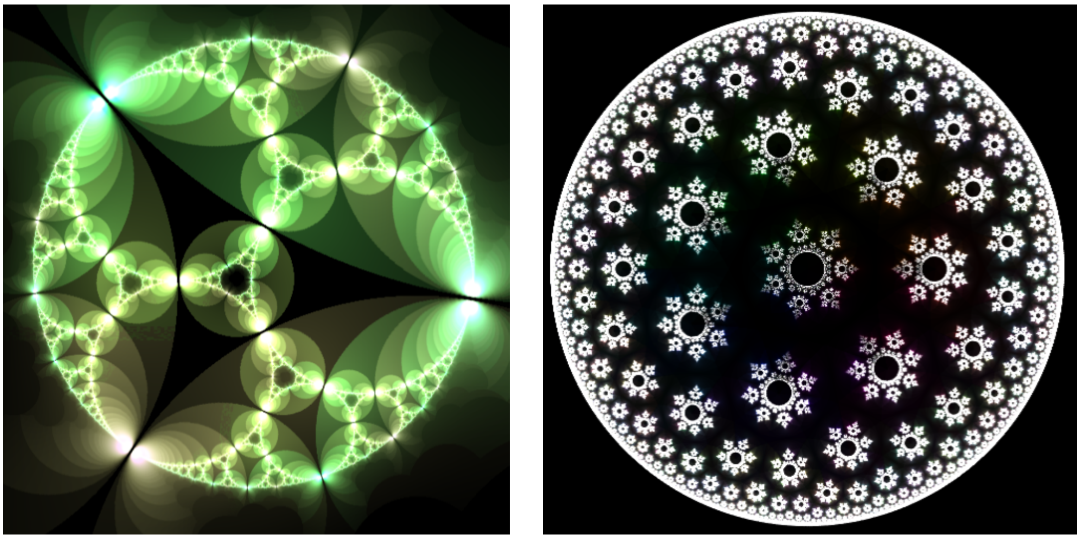


Figure 9: Screenshots of the implementations for fractals generated by circle inversions.

i.e. $C_i := \{z \in \mathbb{C} : |z - a_i| \leq r_i\}$. Then

$$\begin{aligned}\gamma_i : \overline{\mathbb{C} \setminus C_i} &\to C_i \\ z &\mapsto \frac{r_i^2}{\overline{z - c_i}} + c_i\end{aligned}$$

for $i \in [n]$ defines a map that inverts points outside of a circle C_i into the interior of the circle and fixes the boundary. Now we can define $f : \mathcal{H}(\mathbb{C}) \to \mathcal{H}(\mathbb{C})$ pointwise by $f(\{z\}) := \{\gamma_i(z) \in \mathbb{C} : z \in \overline{\mathbb{C} \setminus C_i}\}$. The union of the circles $A = \bigcup_{i=1}^{n} C_i$ is a positively invariant set with respect to f, because the domain of every function γ_i was chosen in such a way that γ_i maps points into C_i only.

Four mutually tangent circles in this setting form the Apollonian gasket as their corresponding limit set: `http://aaron.montag.info/ba/6`.

In the Poincaré disk model of hyperbolic geometry circles that orthogonally intersect the boundary of the disk correspond to lines. Circle inversion at those lines corresponds to hyperbolic reflection. By properly choosing a set of such inversion circles, we can generate a hyperbolic tessellation. Such a fractal was implemented at: `http://aaron.montag.info/ba/7`.

3.2.4 Continuous Escape Time

When considering the images that were generated by our algorithm, then their impression might be enhanced if the steps resulting from the discreteness of the `time` vanished.

This can be done by replacing the constant 1 in the $x \in A \wedge x \notin f(A)$ case of Equation (4) by a continuous function which maps into $[0, 1]$ and interpolates the last step.

A formula for the special case where X is a normed space and $A = \overline{D}_R(0)$ is given in [HPS91]. In this manner the *continuous time* `ctime` $: X \to \mathbb{R}_\infty := \mathbb{R}_{>0} \cup \{\infty\}$ can be

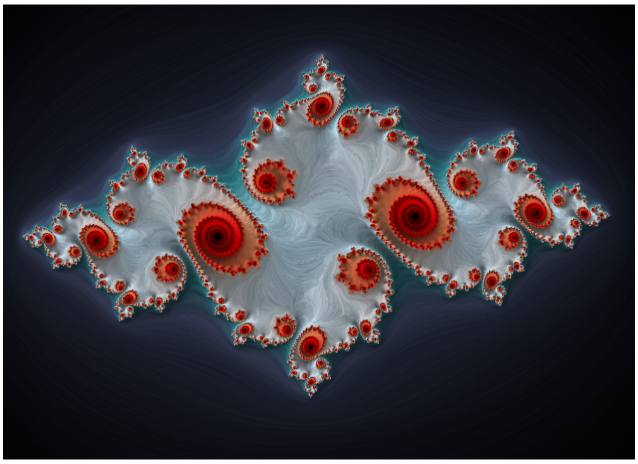
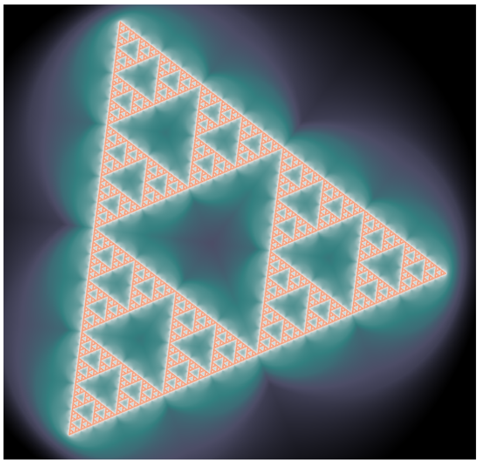
 

Figure 10: Screenshots of the two implementations with continuous escape time.

defined recursively:

$$\texttt{ctime}(x) = \begin{cases} 0 & \text{if } \|x\| > R \\ \max\{\frac{\log(R/\|x\|)}{\log(\|y\|/\|x\|)} \in \mathbb{R}_\infty : y \in f^{-1}(\{x\})\} & \text{if } \|x\| \le R \wedge x \notin f(\overline{D}_R(0)) \\ \max\{\texttt{ctime}(y) + 1 \in \mathbb{R}_\infty : y \in f^{-1}(\{x\})\} & \text{otherwise} \end{cases} \tag{5}$$

The reason for choosing $\frac{\log(R/\|x\|)}{\log(\|y\|/\|x\|)}$ as interpolation is due to the fact that it is the natural continuous extension for `time` of the one-transformation IFS $W(x) = rx$ with $0 < r < 1$, where we have

$$\texttt{time}(x) = \max\{k \in \mathbb{N} : W^{-k}(x) = \frac{x}{r^k} \in \overline{D}_R(0)\} = \left\lfloor \log_r \frac{\|x\|}{R} \right\rfloor_+ = \left\lfloor \frac{\log(R/\|x\|)}{\log(\|\frac{x}{r}\|/\|x\|)} \right\rfloor_+ .$$

According to [HPS91] `ctime` is continuous on $X \setminus \Lambda$, provided that it is applied for a hyperbolic IFSs.

Algorithm 1 can be adapted to visualize `ctime` by replacing line 8 (which is *time* $\leftarrow 1$) by *time* $\leftarrow 0$, and line 14 (which is *newtime* $\leftarrow 1$) by *newtime* $\leftarrow \frac{\log(R/\|x\|)}{\log(\|y\|/\|x\|)}$.

With this continuous escape time, several visualizations are possible. Color palettes might be used to display the wide spectrum of different escape times. One might calculate the gradient of `ctime`, which gives a more complicated recursive formula (that also can be implemented and progressively calculated by our texture based approach). The gradient then can be used to introduce some lightning effects, which then turn some fractal in an alpine landscape. Furthermore a height depended texture can be mapped on this “surface”, which enables the user to distinguish very fine differences of escape times.

We have implemented two optical enhanced examples. One for the filled Julia set and one for the Sierpinski triangle: `http://aaron.montag.info/ba/8` and `http://aaron.montag.info/ba/9`.

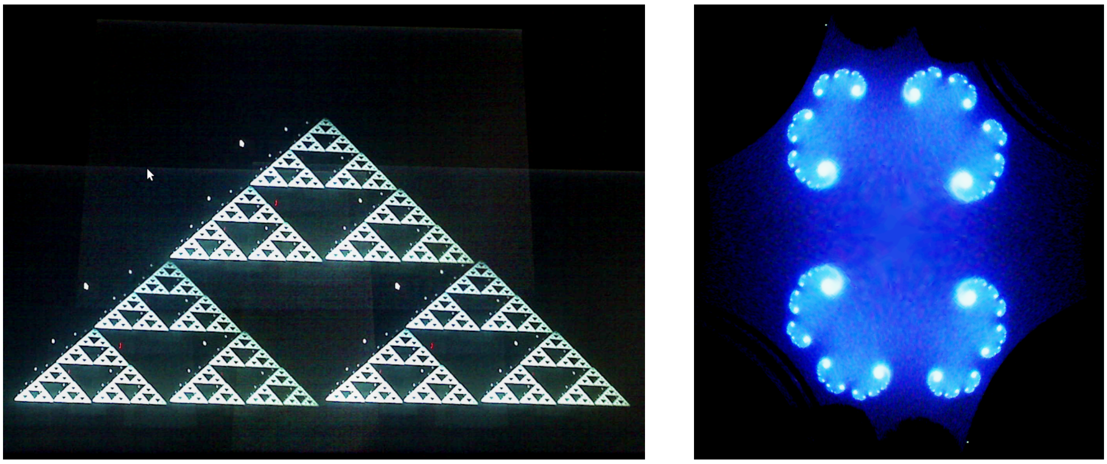


Figure 11: Two fractals generated by analog feedback loops. (Using a webcam and a mirror)

3.2.5 Analog Feedback Loops

Instead of iteratively deforming textures, we can use a analog simple design setup to obtain fractals that are generated by escape time algorithms:

Probably almost everyone once has observed how an “infinite tunnel” becomes visible if one points a camera at a screen which directly displays a live video recorded by the camera. Today it is relatively easy to reproduce this effect. Several technical devices have integrated webcams. For such devices a mirror can be used for moving the screen in the field of view of the camera.

Instead of directly showing the currently recorded image we deform the image on the screen in the sense of Equation (4). For instance, in order to “render” the Sierpinski triangle, we show the recorded image three times simultaneously next to each other, such that each mirrored live record appears optically scaled-down by the factor 2 (compare by Example 3.6). After some time in this setting, the Sierpinski triangle becomes visible (as in Figure 11). An demonstration that accesses the webcam and deformes the recorded video as just described can be found at `http://aaron.montag.info/ba/10`.

A mathematical description can be given in the manner of Definition 3.12: We interpret the screen as a positively invariant set A for a function f which corresponds to the effect of the projective distortion and the artificial deformations on the screen. A pixel on the screen might display a part of the depictured screen. This self-capturing process iterates until something next to the screen is displayed. The number of passages for an pixel x in this loop corresponds to the value of $\mathtt{time}(x)$. Due to physical reasons, the color slightly changes in each passage. Some LCD displays for instance, cause a slight bluish fog during this process. If this natural change in color is not sufficient then one can brighten up the displayed colors on the screen by hand.

We can use this didactic process for any escape time algorithm. In particular it can be used to render Julia sets. The idea is that we color a pixel $z \in \mathbb{C}$ on the screen by the color of the pixel z^2 of the captured image of the camera. Under suitable conditions, the projective distortion caused by the relative position of the camera capturing the screen is (almost) euclidean and therefore corresponds to a function $z \mapsto \alpha z + \beta$ for some $\alpha, \beta \in \mathbb{C}$.

In combination we get $f^{-1} = (z \mapsto \alpha z + \beta) \circ (z \mapsto z^2) = (z \mapsto \alpha z^2 + \beta)$. Using the substitution $S = z \mapsto \alpha z$ it turns out that $f^{-1} = S^{-1} \circ J_{\alpha\beta} \circ S$. Therefore a Julia set becomes visible. The user can regulate its parameter $c = \alpha\beta$ by adjusting the camera. An interactive example is implemented at `http://aaron.montag.info/ba/11`. For results compare Figure 11.

3.3 Hyperbolic Iterated Function Systems with Probabilities

Aside using an escape time algorithm, there is another approach to visualize the limit set of a hyperbolic IFS using feedback loops.With our first approach in Section 3.1.1, when we displayed and deformed sets as black-white images, we had some problem with vanishing fractals. One might consider solving this problem by constantly brightening up non-black pixels. If some set gets scaled down, then the pixels covering this set could be enlightened with the corresponding scaling factor, in order to keep the amount of used “brightness ink” as an invariant.
We did this in order to solve the problem with the vanishing images and obtained very good results. Under the hood, this approach gives some precise mathematics, which was developed first by Hutchinson in [Hut79].
The idea is to work with a sequence of Borel regular measures instead of a sequence of sets. The support of those measures acts in the same way as the sets generated by the Hutchinson operator. Furthermore, we require that the measures always must assign the constant value to the entire space, therefore they will not vanish. The brightness of pixels on the screen will indicate the measure which is assigned to the filled square covered by the pixel.
In Section 3.1 we deformed a set $C \in \mathcal{H}(X)$ by simply taking the image $T(C)$ for a transformation $T : X \to X$. How can we deform a measure?

Definition 3.15 ($\mathcal{M}(X)$, $\mathcal{M}^1(X)$, $T\mu$, hyperbolic IFS with probabilities, Markov Operator)**.** Let (X, d) be a metric space, $\mathcal{M}(X)$ the set of Borel regular measures on X, $\mu \in \mathcal{M}(X)$ a measure and $T : X \to X$ a measurable function. Then we define the measure $T\mu \in \mathcal{M}(X)$ as

$$(T\mu)(A) := \mu(T^{-1}(A))$$

for every A in the σ-algebra over X.
For a given hyperbolic IFS $(X, d, \{w_1, \dots, w_n\})$, we attach to each contraction w_i a probability $p_i \in [0, 1]$ such that $\sum_{i=1}^{N} p_i = 1$ and yield the hyperbolic IFS with probabilities $(X, d, \{(w_1, p_1), \dots, (w_n, p_n)\})$. [8] This IFS with probabilities operates on $\mathcal{M}^1(X) = \{\mu \in \mathcal{M}(X) : \mu(X) = 1\}$, the space of *normalized Borel* measures on X, as follows: The *Markov operator* M associated to the IFS with probabilities is the function

$$\begin{aligned} M : \mathcal{M}^1(X) &\to \mathcal{M}^1(X) \\ \mu &\mapsto \sum_{i=1}^{N} p_i T\mu \,. \end{aligned}$$

[8] The number p_i corresponds to the probability for choosing the contraction w_i in the *random iteration algorithm.* We are not going to cover this algorithm here. For more information, see [Bar12].

Similarly as for the Hutchinson operator, that is a contraction on $(\mathcal{H}(X), h)$, the Markov operator forms a contraction on the space $(\mathcal{M}^1(X), d_H)$ equipped with the *Hutchinson metric* that is defined as

$$d_H(\mu,\nu) := \sup\{\int_X f\,d\mu - \int_X f\,d\nu \mid f: X \to \mathbb{R}, |f(x) - f(y)| \le d(x,y)\,\forall x,y \in X\}$$

for $\mu, \nu \in M^1(X)$. Furthermore if (X,d) is compact, then $(\mathcal{M}^1(X), d_H)$ is compact as well. For a proof of those statements we refer to [Bar12, Chpt. 9]. Accepting this, Banachs fixed-point theorem immediately gives us the following theorem:

Theorem 3.16. *Let (X,d) be a compact space and $(X, d, \{(w_1,p_1),\dots,(w_n,p_n)\})$ a hyperbolic IFS with probabilities and the associated Markov operator $M : \mathcal{M}^1(X) \to \mathcal{M}^1(X)$. Then there exists a unique invariant measure $\nu \in \mathcal{M}^1(X)$ such that*

$$M(\nu) = \nu$$

and furthermore for every $\mu \in \mathcal{M}^1(X)$

$$\lim_{n\to\infty} M^n(\mu) = \nu$$

holds with respect to the Hutchinson distance d_H on $\mathcal{M}^1(X)$.

According to [Hut79, Thm. 4. (ii)] the support of the invariant measure ν equals to the limit set Λ associated to the IFS.

3.3.1 Texture Based Implementation of the Probability Based Approach

We will again use textures to store data. Our aim will be to utilize a texture in order to visualize a normalized Borel measure. Furthermore, we are interested in a method to deform a texture in such a way that corresponds roughly the iterated application of the Markov operator to the corresponding measure.
For practical reasons we will assume that X is a rectangular compact subset of $\mathbb{R}^2$ (or analogously, for $\mathbb{C} \cong \mathbb{R}^2$).
How can we display Borel measures on X on the screen? Let $\epsilon \in \mathbb{R}_{>0}$ be some small number that represents the width of a single pixel. For every $x \in \{(a\epsilon, b\epsilon) \mid a,b \in \mathbb{Z}\} \cap X$ we associate the quadratic pixel

$$P_x = \left[x_1 - \frac{\epsilon}{2}, x_1 + \frac{\epsilon}{2}\right] \times \left[x_2 - \frac{\epsilon}{2}, x_2 + \frac{\epsilon}{2}\right] \subset \mathbb{R}^2\,.$$

For a measure $\mu \in \mathcal{M}(X)$ and a set A in the σ-algebra of X we define the density

$$D_\mu(A) := \frac{\mu(A)}{\mathcal{L}(A)}$$

where $\mathcal{L}(\cdot)$ denotes the two dimensional Lebesgue measure. We will use the brightness of a pixel P_x to indicate the density $D_\mu(P_x)$.

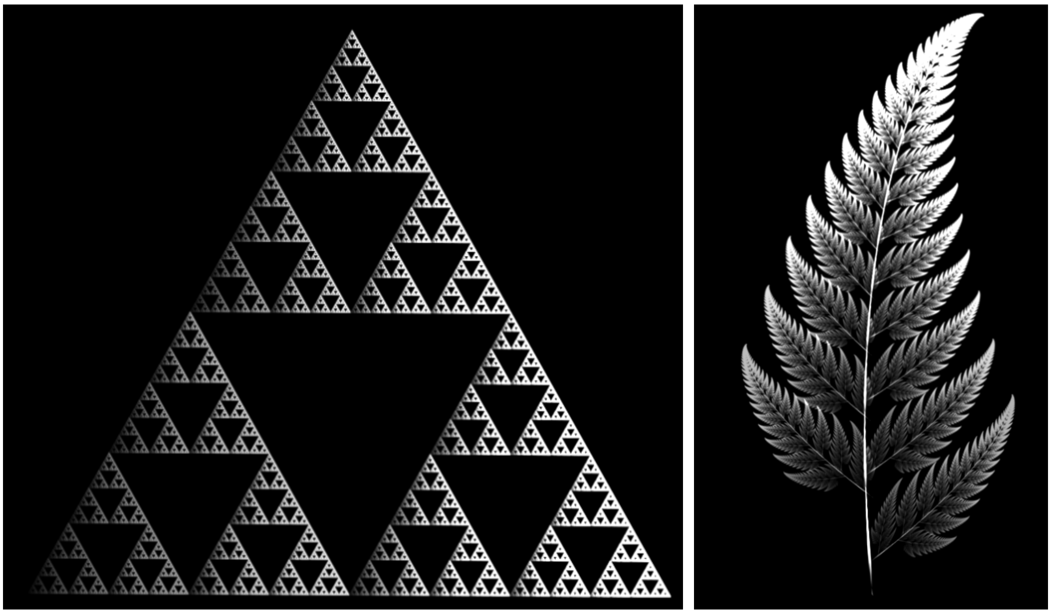


Figure 12: Screenshots of the two implementations using the measure based approach.

What should happen on the screen, if we apply some transformation T to a displayed measure $\mu \in \mathcal{M}(X)$? For simplicity, we will assume that T is a diffeomorphism. Ideally, the value of the pixel P_x should attain the new value $D_{T\mu}(P_x)$. By the definition of $T\mu$ and integration by substitution (See [Els04, "Transformationsformel"]) we yield:

$$D_{T\mu}(P_x) = \frac{\mu(T^{-1}(P_x))}{\mathcal{L}(P_x)} = \frac{\mu(T^{-1}(P_x))}{\mathcal{L}(T^{-1}(P_x))} \frac{\mathcal{L}(T^{-1}(P_x))}{\mathcal{L}(P_x)} = D_\mu(T^{-1}(P_x)) \frac{\int_{P_x} |\det DT^{-1}| \, d\mathcal{L}}{\mathcal{L}(P_x)}$$

The expression $D_\mu(T^{-1}(P_x))$ corresponds to the density of μ over the region $T^{-1}(P_x)$. This value can be approached by taking the average brightness of the corresponding pixels. In OpenGL a method to query such averaged values from a texture is provided by anisotropic texture filtering in combination with automatic generated mipmap textures. A less good approximation of $D_\mu(T^{-1}(P_x))$ is attained by taking the lightness of the pixel that covers the point $T^{-1}(x)$. Assuming that DT is continuous and ϵ sufficient small, the second factor $\frac{\int_{P_x} |\det DT^{-1}| \, d\mathcal{L}}{\mathcal{L}(P_x)}$ can be approached by $\det DT^{-1}(x)$.

Now, suppose that $(X, d, \{(w_1, p_1), \ldots, (w_n, p_n)\})$ is a hyperbolic IFS with probabilities where every contraction w_i is a diffeomorphism. Then the Markov operator of a measure encoded by a texture is the linear combination of transformations of the measure. It is straightforward to approximate the new resulting texture by the formula above.

The following algorithm, which starts with a uniform distribution μ, will approach the invariant measure ν of the associated Markov operator by Theorem 3.16.

Algorithm 2: A method to calculate and visualize the invariant measure ν progressively

```
Initialize two high resolution single channel textures CurrentD and PreviousD of the same size.
foreach pixel P_x on the texture CurrentD do
    CurrentD(P_x) ← c > 0 /* uniform distribution up to a scalar multiple */
end
while program is running do at most (roughly) 30 times a second
    /* The rendering procedure for a single frame */
    Get user input, such as the current mouse coordinate and specified parameters.
    Based on this data calculate the functions w_i^{-1} and Dw_i^{-1} and the probabilities p_i.
    foreach pixel P_x on the texture CurrentD do
        x ← point in X that corresponds to the pixel P_x
        d ← 0
        for i ← 1, 2, ..., n do /* Calculate density of \sum_{i=1}^n p_i w_i(\mu) */
            foreach y ∈ w_i^{-1}(x) do
                p_y ← pixel(coordinate) that corresponds to the point y
                d ← d + p_i · |det Dw_i^{-1}(x)| · PreviousD(p_y)
            end
        end
        CurrentD(P_x) ← d
    end
    Display the texture CurrentD on the screen by, for instance, interpreting high values as bright colors.
    PreviousD ← CurrentD
end
```

Here again, the user might change some parameters at running time. According to Theorem 3.16 we will eventually see an image that is arbitrarily close to the new invariant measure.

An example implementation for the Sierpinski-triangle with the probabilities $p_1 = 0.2$, $p_2 = 0.3$, $p_3 = 0.5$ can be found here: `http://aaron.montag.info/ba/12`.An implementation for Barnsleys farn, which is described in [Bar12], can be found here: `http://aaron.montag.info/ba/13`.

3.4 Groups and Monoids as Languages

In this section we will introduce some notation which is useful in order to generalize our concept of limit sets and will finally lead to another perception of the limit set for hyperbolic IFS.

Let $(X, d, \{w_1, \ldots, w_n\})$ be a hyperbolic IFS. Then the set of all possible finite sequences of concatenations of those transformations $w_1, \ldots, w_n$ plus the identity-transformation $\iota := \mathrm{id}_X$ forms a monoid $(M, \circ)$ which acts on X.[9] Elements of a finitely generated monoid might be described with strings, which we will formalize in the successive definition:

[9] By the phrase $(M, \circ)$ *acts on* X we mean that each element of M can be considered as a map from

Definition 3.17 (adapted from [EPC$^+$92])**.** Let M be a monoid, $\Sigma \subset M$ a finite subset of M. With Σ^* we denote the set of all strings over the alphabet Σ where $\epsilon \in \Sigma^*$ denotes the empty string. $(\Sigma^*, +)$ itself forms a monoid with ϵ as neutral element and where $+$ is the concatenation of strings. In future we will write vw instead of $v + w$, where $v, w \in \Sigma^*$. $|w|$ stands for the length of the word $w \in \Sigma^*$.
$\Sigma^n = \{w \in \Sigma^* : |w| = n\}$ denotes the set of all strings over Σ of length n.
By interpreting concatenation in Σ^* as multiplication in M the map $\pi : \Sigma^* \to M$ forms a monoid homomorphism. In this sense we are constrained to set $\pi(\epsilon) = \iota$.
Now, we say Σ generates the monoid M *as a semigroup*, or Σ is the set of semigroup generators for M, iff the interpretation map $\pi : \Sigma^* \to M$ is surjective.[10]
All the definitions can also be applied for a group G as well, because a group is, in some sense nothing but kind of a special monoid. Note that here the term $\Sigma \subset G$ generates G *as a semigroup* differs from the standard term $\Sigma \subset G$ generates G, which is equivalent to the term $\Sigma \cup \Sigma^{-1}$ generates G *as a semigroup*.

Example 3.18. The group $(\mathbb{Z}, +)$ is generated as a semigroup by the alphabet $\Sigma = \{\texttt{+1}, \texttt{-1}\} \subset \mathbb{Z}$. Therefore $\texttt{+1+1-1} \in \Sigma^*$ and $|\texttt{+1+1-1}| = 3$ and $\pi(\texttt{+1+1-1}) = 1+1-1 = 1 \in \mathbb{Z}$.

Example 3.19. $\Sigma = \{\mathtt{w}_1, \ldots, \mathtt{w}_n\}$ is the set of semigroup generates for the monoid associated to the IFS $(X, d, \{w_1, \ldots, w_n\})$.

3.5 An Alternative Description of the Limit Set

So far we encountered the term limit set only for hyperbolic Iterated Function Systems. The goal of this section is to understand of what the limit set explicitly consists of. In the next section we will use these insights to generalize this concept in order to investigate the limit sets of Kleinian Groups.
We will start with a proposition that uses the definitions we introduced in the last section.

Proposition 3.20. *Let M denote the monoid associated to the IFS $(X, d, \{w_1, \ldots, w_n\})$ and $\Sigma = \{\mathtt{w}_1, \ldots, \mathtt{w}_n\}$. Then one might interpret the iterated Hutchinson-operator as follows:*

$$W^n(C) = \bigcup_{w \in \Sigma^n} \pi(w)(C)$$

Proof by induction on n. For the base case $n = 0$, the union is over the empty string ϵ, which correspondents to the identity transformation. This is consistent with $W^0(C) = C$. For the inductive step we will assume that $W^n(C) = \bigcup_{w \in \Sigma^n} \pi(w)(C)$ and now apply the

---

X to itself where the following conditions hold for all $x \in X, a, b \in M$:

$$\iota(x) = x$$
$$a(b(x)) = (a \circ b)(x)$$

[10] Actually the term Σ generates M as monoid would be more suitable, but we will comply with the conventional notation.

Hutchinson operator W on $W^n(C)$. Then with the homomorphism property of π it turns out after some rearrangement, that $W^{n+1}(C)$ can be written in the same manner:

$$\begin{aligned} W^{n+1}(C) = W(W^n(C)) &= \bigcup_{i=1}^{n} w_i \bigcup_{w\in\Sigma^n} \pi(w)(C) = \bigcup_{\sigma\in\Sigma} \pi(\sigma) \bigcup_{w\in\Sigma^n} \pi(w)(C) \\ &= \bigcup_{\sigma\in\Sigma} \bigcup_{w\in\Sigma^n} \pi(\sigma w)(C) = \bigcup_{w\in\Sigma^{n+1}} \pi(w)(C) \end{aligned}$$

□

Now we have got the formal machinery to give an alternative description of the limit set Λ for a hyperbolic IFS. In order to do so, we will take a closer look at the orbits that were induced by the action of M.

Theorem 3.21. *Let (X, d) be a complete metric space that contains at least two points, $(X, d, \{w_1, \ldots, w_n\})$ a hyperbolic IFS of injective transformations with the associated monoid M. Then the limit set Λ of $(X, d, \{w_1, \ldots, w_n\})$ (as it is defined in Corollary 3.5 by iterating the Hutchinson operator on an arbitrary non-empty compact set) equals to*

$$\Lambda(M) := \{z \in X : \textit{there exists } x \in X \textit{ and a sequence } (m_n)_{n\in\mathbb{N}} \textit{ of pairwise disjoint monoid elements in } M \textit{ such that } \lim_{n\to\infty} m_n x = z\},$$

which contains the accumulation points of the occurring orbits of M acting on X.

We need to prove a technical lemma first.

Lemma 3.22. *Suppose we are in the setting of Theorem 3.21. Let $\Sigma = \{\mathtt{w}_1, \ldots, \mathtt{w}_n\}$, $v_n \in \Sigma^n$ a sequence of words of increasing lengths. Then there is a subsequence $(\pi(v_{n_k}))_{k\in\mathbb{N}}$ of pairwise disjoint transformations.*

Proof. Let $x, y \in X$ be two distinct points. Thus $d(x, y) > 0$. Since M consists of injective transformations only (it is generated by injective transformations), we conclude that $\pi(v_n)x \neq \pi(v_n)y$ for every n, or equivalently, in terms of the metric:

$$\forall n \in N : d(\pi(v_n)x, \pi(v_n)y) > 0$$

On the other hand, $v_n \in \Sigma^n$ gives us $d(\pi(v_n)x, \pi(v_n)y) < L^n d(x, y)$, where $L := \max_{i\in[n]} L_i < 1$ and the L_is are the Lipschitz constants for w_i respectively. Therefore $\lim_{n\to\infty} d(\pi(v_n)x, \pi(v_n)y) = 0$.
Consequently, $(d(\pi(v_n)x, \pi(v_n)y))_{n\in\mathbb{N}}$ is a zero sequence of non vanishing values. Hence there must be a subsequence $(n_k)_{k\in\mathbb{N}}$ such that the values of the sequence $(d(\pi(v_{n_k})x, \pi(v_{n_k})y))_{k\in\mathbb{N}}$ are all different. Therefore also the sequence $(\pi(v_{n_k}))_{k\in\mathbb{N}}$ consists of pairwise disjoint transformations. □

Proof of Theorem 3.21. Let $\Sigma = \{\mathtt{w}_1, \ldots, \mathtt{w}_n\} \subset M$ an alphabet, which generates M as a semigroup. For this proof we will use extensively the explicit characterization of limits of convergent sequences in $\mathcal{H}(X)$ (see Theorem 3.4).

Assert that $z \in \Lambda$. Our job is to show that $z \in \Lambda(M)$. Let $x \in X$ be some arbitrary point. Obviously $\{x\} \in \mathcal{H}(X)$ and henceforth by Corollary 3.5 $\lim_{n\to\infty} W^n\{x\} = \Lambda$. Now the characterization of limits in Theorem 3.4 gives us the following equivalence

$$\begin{aligned} z \in \Lambda &\Leftrightarrow z \in \lim_{n\to\infty} W^n\{x\} \\ &\Leftrightarrow \text{there is a convergent sequence } z_n \in W^n\{x\} \text{ with } \lim_{n\to\infty} z_n = z \\ &\Leftrightarrow \text{there is a sequence of words } v_n \in \Sigma^n \text{ such that } \lim_{n\to\infty} \pi(v_n)(x) = z \end{aligned}$$

The just proven Lemma 3.22 allows us to take some subsequence $v_{n_k} \in \Sigma^{n_k}$ from $(v_n)_{n\in\mathbb{N}}$ for which the interpretations $\pi(v_{n_k}) \in M$ are pairwise distinct. The limit $\lim_{k\to\infty} \pi(v_{n_k})(x) = z$ is preserved, therefore with the starting point x and the sequence $\pi(v_{n_k})$ we fulfill the requirement for $z \in \Lambda(M)$.
For the other implication let us assume $z \in \Lambda(M)$, i.e. let $x \in X$ and $(m_n)_{n\in\mathbb{N}}$ be a sequence of pairwise disjoint elements in M with $\lim_{n\to\infty} m_n x = z$. Now choose $v_n \in \pi^{-1}(m_n)$ as some sequence of words representing those transformations. Since the monoid elements m_n are pairwise disjoint, whilst the sets Σ^n are finite for every $n \in \mathbb{N}$, it turns out that $\lim_{n\to\infty} |v_n| = \infty$. This implies

$$\lim_{n\to\infty} W^{|v_n|}\{x\} = \lim_{n\to\infty} W^n\{x\} = \Lambda\,.$$

The explicit characterization of the Hausdorff limit $\lim_{n\to\infty} W^{|v_n|}\{x\}$ given in Theorem 3.4 tells us that one can reason

$$z = \lim_{n\to\infty} m_n x \in \lim_{n\to\infty} W^{|v_n|}\{x\} = \Lambda$$

from $m_n x \in W^{|v_n|}\{x\}$. □

Hereby we have given some alternative description of the limit set of non-trivial hyperbolic IFS with injective transformations: It also is the set of accumulation points of the orbits of M.
The requirement of non-injectivity was asserted for this characterization, in order to exclude any IFS that induces finite monids, for which the definition $\Lambda(M)$ always gives us the empty set. For instance, consider the IFS $(X, d, \{w_1\})$ with the non-injective map $w_1 : X \to X, x \mapsto z$ mapping everything to some $z \in X$. This IFS generated the finite monoid $M = \{\iota, w_1\}$, hence $\Lambda(M) = \varnothing$, but $\Lambda = \lim_{n\to\infty} W^n(\{z\}) = \{z\}$.

4 Kleinian Groups

In the last chapter we have studied hyperbolic Iterated Function Systems, where we demanded every transformation to be a contraction.
We will investigate groups of Möbius transformations, that act on $\hat{\mathbb{C}}$, and give an algorithm to draw corresponding limit sets by converging images. This will be harder than it was for IFSs because taking the Hutchinson operator $W = C \mapsto \bigcup_{\sigma \in \Sigma} \pi(\sigma)C$ on an alphabet of transformations and semigroup generators Σ for a group G will not work anymore.

4.1 Mathematical Fundamentals of Kleinian Groups

We will first define the domain Möbius transformations are acting on, the extended complex numbers, or the Riemann sphere:

Definition 4.1 (The extended complex numbers $\hat{\mathbb{C}}$ and their metric d)**.** We set

$$\hat{\mathbb{C}} = \mathbb{C} \cup \{\infty\}$$

and extend the arithmetic of $\mathbb{C}$ by setting $z + \infty = \infty$, $z \cdot \infty = \infty$, $\frac{z}{\infty} = 0$ and $\frac{z}{0} = \infty$ for all $z \in \mathbb{C}$.

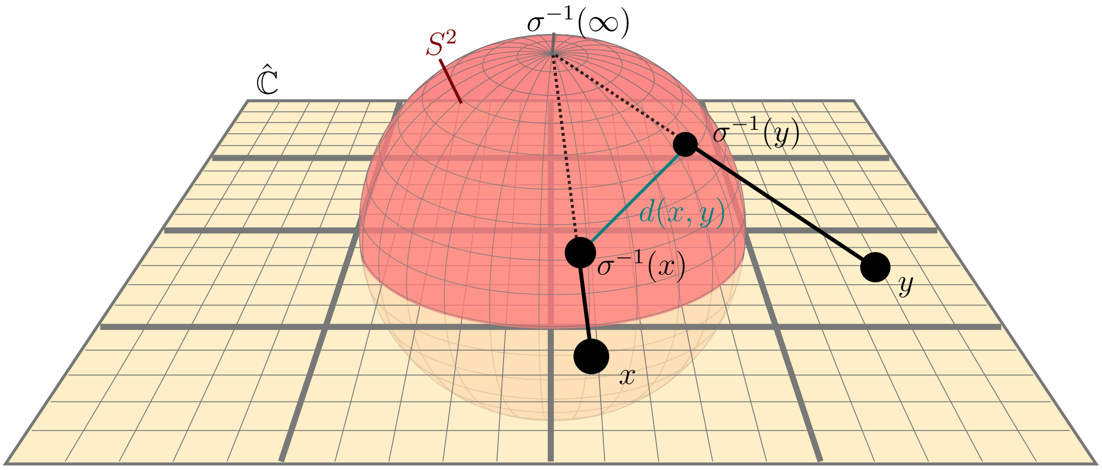


Figure 13: The Riemann sphere and its stereographic projection to $\hat{\mathbb{C}}$ (Actually the finite plane + a single point at infinity). The distance $d(x, y)$ for $x, y \in \hat{\mathbb{C}}$ is delineated in teal.

In the manner of [HH99] we equip $\hat{\mathbb{C}}$ with a distance function in order to make it a metric space. Using stereographic projection, we pull back the metric from the Riemann sphere $S^2 = \{(z, y) \in \mathbb{C} \times \mathbb{R} \mid |z| + y^2 = 1\}$. The stereographic projection $\sigma : S^2 \to \hat{\mathbb{C}}$ is a bijective map with

$$\sigma(z, y) = \begin{cases} \frac{z}{1-y} & \text{if } (z, y) \neq (0, 1) \\ \infty & \text{if } (z, y) = (0, 1) \end{cases} \qquad \sigma^{-1}(z) = \begin{cases} \left(\frac{2z}{|z|^2+1}, \frac{|z|^2-1}{|z|^2+1}\right) & \text{if } z \in \mathbb{C} \\ (0, 1) & \text{if } z = \infty \end{cases}$$

For $x, y \in \hat{\mathbb{C}}$ we set

$$d(x, y) := |\sigma^{-1}(x) - \sigma^{-1}(y)| \, .$$

Let $(a_n)_{n\in\mathbb{N}}$ and $(b_n)_{n\in\mathbb{N}}$ sequences in $\mathbb{C}$. Clearly, if $\lim_{n\to\infty}|a_n-b_n| = 0$, then $\lim_{n\to\infty} d(a_n, b_n) = 0$. On the other hand, if $\lim_{n\to\infty}|z_n| = \infty$, then $\lim_{n\to\infty} d(z_n,\infty) = 0$. In particular, our term of convergence with respect to the metric induced by the standard euclidean norm on $\mathbb{C}$ is extended by the convergence with respect to d.

Remark 4.2. Every sequence $(z_n)_{n\in\mathbb{N}}$ in $\hat{\mathbb{C}}$ has a convergent subsequence. In other words, every closed set in $\hat{\mathbb{C}}$ is sequentially compact.

Proof. Without loss of generality we may assume that the sequence has no member z_n s.t. $z_n = \infty$. If there are infinitely members becoming ∞ then ∞ is an accumulation point. Otherwise we can consider the sequence after a finite number of occurring ∞s. Suppose $(z_n)_{n\in\mathbb{N}}$ is bounded, then by the Bolzano-Weierstraß theorem it has an accumulation point. If it is not bounded then it has a subsequence $(z_{n_k})_{k\in\mathbb{N}}$ such that $\lim_{k\to\infty}|z_{n_k}| = \infty$, which is by definition equivalent to $\lim_{k\to\infty} z_{n_k} = \infty$. □

Definition 4.3 (Möbius Transformations)**.** Let $\gamma = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \in \mathrm{SL}(2,\mathbb{C})$, i.e. $\det\gamma = ac - bd = 1$. Then the function $M_\gamma : \hat{\mathbb{C}} \to \hat{\mathbb{C}}$ with

$$M_\gamma(z) = \frac{az+b}{cz+d}$$

and $M_\gamma(\infty) = \frac{a}{c}$ is called a *Möbius Transformation* with the associated matrix γ.

Remark 4.4. With the notation as above, we have $M_\gamma = M_{\alpha\gamma}$ for any $\alpha \in \mathbb{C}^\times$, as the numerator and denumerator can be reduced by the same number. So if γ is only a regular matrix in $\mathrm{GL}(2,\mathbb{C})$, i.e. $\det\gamma = ac-bd \neq 0$, then the function M_γ can also be considered as a Möbius transformation, with the associated matrix $\frac{1}{\sqrt{\det\gamma}}\gamma \in \mathrm{SL}(2,\mathbb{C})$. In the successive calculations we always assume that $\det\gamma = ac - bd = 1$.

Lemma 4.5. *Möbius transformations act biholomorphic on* $\hat{\mathbb{C}}$*. Furthermore, for* $\gamma_1,\gamma_2 \in \mathrm{GL}(2,\mathbb{C})$ *we have the property* $M_{\gamma_1}\circ M_{\gamma_2} = M_{\gamma_1\cdot\gamma_2}$*. For* $I := \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ *one obtains* $M_I = id|_{\hat{\mathbb{C}}}$*. If* $\gamma = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \in \mathrm{SL}(2,\mathbb{C})$ *then we have* $M_\gamma^{-1} = M_{\gamma^{-1}}$ *with* $\gamma^{-1} = \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$*. In particular, Möbius transformations form a subgroup of* $\mathrm{Aut}(\hat{\mathbb{C}})$*.*

Proof. An elegant proof using homogeneous coordinates is given in [RG11, Ch. 7]. Their biholomorphic nature is proven in [HH99]. It is worth to remark, that every biholomorphic automorphism on $\hat{\mathbb{C}}$ is a Möbius transformation. □

From now on $Tr\gamma = a + d$ denotes the trace for the matrix $\gamma = (a\ b|c\ d) \in \mathrm{SL}(2,\mathbb{C})$. With the term "$M_\gamma$ is conjugated to the Möbius transformation S" we mean that there is another Möbius transformation T such that $M_\gamma = T\circ S\circ T^{-1}$. Note that the trace of the representing matrices remains invariant under conjugation.

Definition-Lemma 4.6 (Classification of Möbius Transformations)**.** *Any Möbius transformation* M_γ *different from the identity can be classified as*

parabolic *if* $\operatorname{Tr}\gamma \in \{-2, 2\}$. *It has one fixed point and is conjugated to the Möbius transformation* $z \mapsto z + a$ *with* $a \in \mathbb{C} \setminus \{0\}$.

elliptic *if* $\operatorname{Tr}\gamma \in (-2, 2)$. *It has two fixed points and it is conjugated to the Möbius transformation* $z \mapsto kz$ *with* $k \in \mathbb{C}, |k| = 1$.

loxodromic *if* $\operatorname{Tr}\gamma \in \mathbb{C} \setminus [-2, 2]$. *It has an attracting and repulsive fixed point, where every point in* $\hat{\mathbb{C}}$ *but the repulsive fixed point is attracted to the attracting fixed point by iterating* M_γ. *It is conjugate to the Möbius transformation* $z \mapsto kz$ *with* $k \in \mathbb{C} \setminus \mathbb{R}, |k| > 1$. *Furthermore, we call* M_γ ***hyperbolic****, if* $\operatorname{Tr}\gamma \in \mathbb{R} \setminus [-2, 2]$. *It is a special loxodromic transformation. It is conjugate to the Möbius transformation* $z \mapsto kz$ *with* $k \in \mathbb{R}, k > 1$.

Proof. See [MSW02] or consider the 2×2 Jordan normal form of γ. $\square$

By Lemma 4.5, any group of Möbius transformations operates on $\hat{\mathbb{C}}$ in a natural way. The accumulation points of the occurring orbits are an interesting object to study.
We will basically use the same property as we have developed for hyperbolic IFSs in Theorem 3.21 in order to define the limit set of a group of Möbius transformations:

Definition 4.7 (Limit set, Kleinian group)**.** Let G be a group of Möbius transformations. The *limit set*

$$\Lambda(G) := \{z \in \hat{\mathbb{C}} \mid \text{there exists } w \in \hat{\mathbb{C}} \text{ and a sequence } (g_n)_{n\in\mathbb{N}} \text{ of pairwise disjoint group elements in } G \text{ such that } \lim_{n\to\infty} g_n w = z\}$$

of G is defined as the set of all points in $\hat{\mathbb{C}}$ where the orbit of some $w \in \hat{\mathbb{C}}$ has an accumulation point. The set $\Omega(G) = \hat{\mathbb{C}} \setminus \Lambda(G)$ is called *domain of discontinuity* or the *ordinary set*. We will call points in $\Omega(G)$ *ordinary*.
Those groups G for which $\Omega(G)$ is non-empty are called *Kleinian groups*.
With $\Gamma(G) = \{\gamma \in \mathrm{SL}(2, \mathbb{C}) \mid M_\gamma \in G\}$ we will denote the set of the associated matrices to G.

Note that by definition finite groups G have an empty limit set $\Lambda(G)$. A Kleinian group that is generated by a single loxodromic or parabolic Möbius transformation has its fixed point(s) as limit set (Compare with classification in Definition-Lemma 4.6). A two generator Kleinian group can possibly have rich geometric structures as limit set. An example is depicted in Figure 14. Here we have chosen the two generators by *grandma's recipe* from [MSW02] with the parameters $t_a = 1.9 + 0.1i$ and $t_b = 2.0$. [11]
Our main goal is to develop an algorithm that can, given a finitely generated group G of Möbius transformations, produce a sequence of compact sets that converges with respect to the Hausdorff metric to the limit set $\Lambda(G)$.
First we prove some properties which give some insights into Kleinian groups and are needed afterwards for our goal. Those ideas and their proofs came from or were adapted from [Leh64, Ch. III]

[11] Given two complex parameters t_a and t_b, grandma's recipe computes two Möbius transformations a and b with the traces t_a and t_b respectively. They are chosen in such a way that the commutator $abAB$ has trace -2 (A and B are the inverse transformations of a and b). Therefore $abAB$ is parabolic. Furthermore $abAB$ has the single fixed point 1 and $aBAb$ has the single fixed point -1.

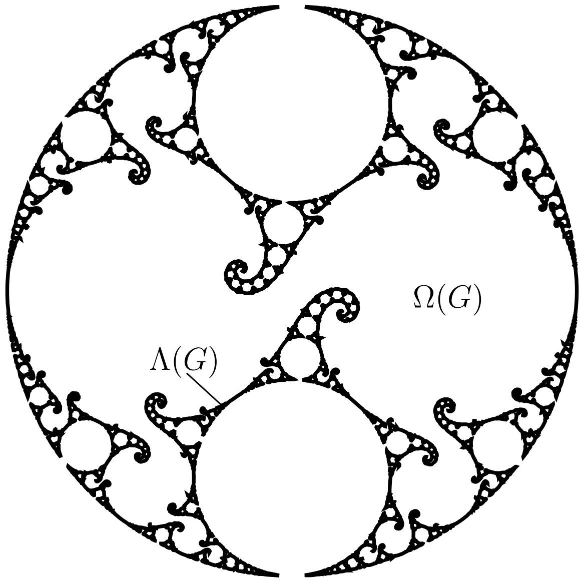


Figure 14: The limit set for a Kleinian group generated by two Möbius transformations.

What happens if we apply some Möbius transformation to the limit set/ordinary set of a Kleinian group? It turns out that the transformed set is the limit set/ordinary set of the group conjugated by the transformation:

Proposition 4.8 (Conjugation of Kleinian Groups, Invariance of $\Lambda(G)$ and $\Omega(G)$ under operations of G)**.** *Let G be a Kleinian group, $T : \hat{\mathbb{C}} \to \hat{\mathbb{C}}$ some Möbius transformation. Then*

$$T\Lambda(G) = \Lambda(TGT^{-1}) \qquad \text{and} \qquad T\Omega(G) = \Omega(TGT^{-1})\,. \tag{6}$$

In particular, if $T \in G$, then

$$T\Lambda(G) = \Lambda(G) \qquad \text{and} \qquad T\Omega(G) = \Omega(G)\,. \tag{7}$$

Proof. First we show that $T\Lambda(G) \subset \Lambda(TGT^{-1})$:
Suppose $z \in T\Lambda(G)$, or in other words, there exists $w \in \hat{\mathbb{C}}$ and a sequence $(g_n)_{n\in\mathbb{N}}$ of pairwise disjoint group elements in G such that $T(\lim_{n\to\infty} g_n w) = z$. From the continuity of T on $\hat{\mathbb{C}}$ follows $z = \lim_{n\to\infty} T(g_n w) = \lim_{n\to\infty}(Tg_nT^{-1})(Tw)$. The sequence $(Tg_nT^{-1})_{n\in\mathbb{N}}$ remains consisting of distinct elements. Therefore $z \in \Lambda(TGT^{-1})$.
The just proven inclusion is true for any Möbius transformation and any group of Möbius transformations, in particular for the transformation T^{-1} and the group TGT^{-1}. Hence $T^{-1}\Lambda(TGT^{-1}) \subset \Lambda\left(T^{-1}(TGT^{-1})T\right) = \Lambda(G)$. Since T is bijective, we apply it to both sides and get $\Lambda(TGT^{-1}) \subset T\Lambda(G)$. So the first equality in (6) is proven.
The equality for the ordinary set in (6) immediately follows from the bijective nature of T:

$$T\Omega(G) = T(\hat{\mathbb{C}} \setminus \Lambda(G)) = T(\hat{\mathbb{C}}) \setminus T\Lambda(G) = \hat{\mathbb{C}} \setminus \Lambda(TGT^{-1}) = \Omega(TGT^{-1})$$

G is a normal subgroup of G itself. In other words, $G = TGT^{-1}$ for any $T \in G$. This implies Equation (7) from Equation (6). $\square$

Another property, we want to show, is the discreteness of a Kleinian group.

Definition 4.9 (discrete set)**.** Let (X, d) be a metric space. A set $S \subset X$ is called *discrete* if there is no $s \in S$ such that there is a sequence $(s_n)_{n\in\mathbb{N}}$ of pairwise disjoint elements in S converging to s, or in other words, every point of S is isolated.

In order to speak of converging matrices in $\mathrm{SL}(2,\mathbb{C})$, we will identify the $\mathbb{C}^{2\times 2}$ canonically with the $\mathbb{C}^4$ equipped with the euclidean metric. Then a sequence of matrices converges if and only if every component converges.

Lemma 4.10. *The set of the associated matrices* $\Gamma(G) = \{\gamma \in \mathrm{SL}(2,\mathbb{C}) \mid M_\gamma \in G\}$ *of the transformations of a Kleinian group* G *has no accumulation points in* $\mathbb{C}^{2\times 2}$*.*

Proof. Assume that there is a sequence $(\gamma_n)_{n\in\mathbb{N}}$ of pairwise disjoint matrices with $\gamma_n = \begin{pmatrix} a_n & b_n \\ c_n & d_n \end{pmatrix} \in \Gamma(G)$ converging to $\gamma = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \in \mathbb{C}^{2\times 2}$. As a consequence

$$1 = \lim_{n\to\infty} \det \begin{pmatrix} a_n & b_n \\ c_n & d_n \end{pmatrix} = \lim_{n\to\infty} a_n d_n - b_n c_n = ad - bc = \det \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

holds. Therefore γ is invertible and

$$\lim_{n\to\infty} \gamma_n \cdot \gamma^{-1} = \lim_{n\to\infty} \begin{pmatrix} a_n & b_n \\ c_n & d_n \end{pmatrix} \cdot \begin{pmatrix} d & -b \\ -c & a \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} .$$

Thus for every $z \in \hat{\mathbb{C}}$ we obtain

$$\lim_{n\to\infty} M_{\gamma_n}(M_{\gamma^{-1}} z) = \lim_{n\to\infty} M_{\gamma_n \cdot \gamma^{-1}} z = z \,,$$

thus $z \in \Lambda(G)$. The last equality follows from the continuity of the operations as multiplication, summation and division, which were used to built Möbius transformations. The result $\Lambda(G) = \hat{\mathbb{C}}$, or $\Omega(G) = \hat{\mathbb{C}} \setminus \Lambda(G) = \varnothing$, contradicts the property of G being Kleinian. $\square$

From Lemma 4.10 directly follows:

Corollary 4.11. $\Gamma(G)$ *is discrete.*

The next two statements require $\infty \in \Omega(G)$, which makes calculations very handy. The idea for using this assumption came from [Leh64]. In fact, this requirement is not a real restriction: As we will show, given some Kleinian group G with $w \in \Omega(G) \neq \varnothing$, then one can conjugate all the transformations in G by some Möbius transformation in order to move $\Lambda(G)$ away from ∞.

Lemma 4.12. *Let* G *be a Kleinian Group with* $\infty \in \Omega(G)$*. Consider a sequence* $(g_n)_{n\in\mathbb{N}}$ *of* pairwise disjoint *elements of* G*. Let* $\gamma_n = \begin{pmatrix} a_n & b_n \\ c_n & d_n \end{pmatrix} \in \mathrm{SL}(2,\mathbb{C})$ *such that* $M_{\gamma_n} = g_n$*. Then*

$$\lim_{n\to\infty} |c_n| = \infty \,.$$

Proof. Our goal is to derive a contradiction by showing that the matrices accumulate at some point.
Assume that $|c_n|$ does not converge to ∞. Then the bounded sequence $(c_n)_{n\in\mathbb{N}}$ has a convergent subsequence. By taking the subsequence we assume without loss of generality that $\lim_{n\to\infty} c_n = c \in \mathbb{C}$.
Because of the property $\infty \notin \Lambda(G)$ we conclude that the three sequences

$$g_n(\infty) = \frac{a_n}{c_n} \qquad g_n^{-1}(\infty) = \frac{d_n}{-c_n} \qquad g_n(0) = \frac{b_n}{d_n}$$

in $\hat{\mathbb{C}}$ have no subsequence converging to ∞.
Thus all of these sequences will stay in $\mathbb{C}$ (and are bounded) after some finite index. So by iteratively extracting converging subsequences we may choose a sequence $(n_k)_{k\in\mathbb{N}}$ such that

$$\lim_{k\to\infty} \frac{a_{n_k}}{c_{n_k}} = \alpha_1 \qquad \lim_{k\to\infty} \frac{d_{n_k}}{-c_{n_k}} = \alpha_2 \qquad \lim_{k\to\infty} \frac{b_{n_k}}{d_{n_k}} = \alpha_3$$

where $\alpha_1, \alpha_2, \alpha_3 \in \mathbb{C}$.
It turns out that the matrices $\gamma_{n_k} \in \Gamma(G)$ converge in $\mathbb{C}^{2\times 2}$:

$$\lim_{k\to\infty} c_{n_k} = c \qquad \lim_{k\to\infty} a_{n_k} = \lim_{k\to\infty} c_{n_k} \cdot \frac{a_{n_k}}{c_{n_k}} = c \cdot \alpha_1$$

$$\lim_{k\to\infty} d_{n_k} = \lim_{k\to\infty} -c_{n_k} \cdot \frac{d_{n_k}}{-c_{n_k}} = -c \cdot \alpha_2 \qquad \lim_{k\to\infty} b_{n_k} = \lim_{k\to\infty} d_{n_k} \cdot \frac{b_{n_k}}{d_{n_k}} = -c \cdot \alpha_2 \cdot \alpha_3$$

This contradicts Lemma 4.10, which states that $\Gamma(G)$ has not any accumulation point. □

The next proposition will be a useful tool for us determining convergence. It was adapted from a proof in [Leh64].

Proposition 4.13. *Let G be a Kleinian group with $\infty \in \Omega(G)$, $(z_n)_{n\in\mathbb{N}} \subset \hat{\mathbb{C}}$ a convergent sequence and $(g_n)_{n\in\mathbb{N}} \subset G$ a sequence of pairwise disjoint Möbius transformations. Then either*

$$\lim_{n\to\infty} |g_n\infty - g_n z_n| = 0 \tag{8}$$

or

$$z := \lim_{n\to\infty} z_n \in \Lambda(G) \tag{9}$$

or both hold.

Proof. Let $\gamma_n = \begin{pmatrix} a_n & b_n \\ c_n & d_n \end{pmatrix} \in \mathrm{SL}(2,\mathbb{C})$ such that $M_{\gamma_n} = g_n$.
Since $\infty \in \Lambda(G)$ we know that $\lim_{n\to\infty} |c_n| = \infty$ by Lemma 4.12.
Futhermore, one might write

$$|g_n\infty - g_n z_n| = \left|\frac{a_n}{c_n} - \frac{a_n z_n + b_n}{c_n z_n + d_n}\right| = \left|\frac{a_n c_n z_n + a_n d_n - a_n c_n z_n - c_n d_n}{c_n(c_n z_n + d_n)}\right| = \frac{1}{|c_n||c_n z_n + d_n|}\,. \tag{10}$$

1st case: The sequence $|c_n z_n + d_n|$ has no accumulation point at 0 So there is an $N \in \mathbb{N}, \epsilon \in \mathbb{R}_{>0}$ such that for all $n \geq N$ we have $|c_n z_n + d_n| \geq \epsilon$. With Equation (10) we yield

$$\lim_{n\to\infty} |g_n\infty - g_n z_n| \leq \lim_{n\to\infty} \frac{1}{\epsilon |c_n|} = 0$$

as $\lim_{n\to\infty} |c_n| = \infty$ and Equation (8) is proven.

2nd case: there exists a sequence $(k_n)_{n\in\mathbb{N}}$ such that $\lim_{k\to\infty} |c_{n_k} z_{n_k} + d_{n_k}| = 0$. Then we are able to conclude Equation (9) as follows:

$$\lim_{k\to\infty} |z - g_{n_k}^{-1}\infty| \leq \lim_{k\to\infty} |z - z_{n_k}| + \lim_{k\to\infty} |z_{n_k} + \frac{d_{n_k}}{-c_{n_k}}| = 0 + \lim_{k\to\infty} \left| \frac{c_{n_k} z_{n_k} + d_{n_k}}{c_{n_k}} \right| = 0$$

Thus $z \in \Lambda(G)$.

□

Now by dropping the requirement $\infty \in \Omega(G)$, we will generalize this proposition using the metric d on $\hat{\mathbb{C}}$, which will lead us to a precious "universal property":

Lemma 4.14. *Let G be a Kleinian group with $w \in \Omega(G)$ and $(z_n)_{n\in\mathbb{N}} \subset \hat{\mathbb{C}}$ a convergent sequence with limit $z \in \Omega(G)$ and $(g_n)_{n\in\mathbb{N}} \subset G$ a sequence of pairwise disjoint Möbius transformations. Then*

$$\lim_{n\to\infty} d(g_n w, g_n z_n) = 0 \tag{11}$$

Proof. Conjugate the group G with a distance preserving[12] Möbius transformation $T : \hat{\mathbb{C}} \to \hat{\mathbb{C}}$ with $Tw = \infty$. Such a distance preserving Möbius transformation exists: Just take $T = \sigma \circ R \circ \sigma^{-1}$ where $R : S^2 \to S^2$ simply rotates the Riemann sphere such that $\sigma^{-1}w$ is moved to $\sigma^{-1}\infty$. According to [Nee11] T forms a Möbius transformation.
By Proposition 4.8 $\Omega(TGT^{-1}) = T\Omega(G)$ holds. According to that $\infty = Tw \in \Omega(TGT^{-1})$ and by the continuity of T $\lim_{n\to\infty} Tz_n = Tz \in \Omega(TGT^{-1})$. So we are in the first case of Proposition 4.13 for the Kleinian group TGT^{-1} and get

$$\lim_{n\to\infty} |(Tg_nT^{-1})\infty - (Tg_nT^{-1})(Tz_n)| = 0\,,$$

hence

$$\lim_{n\to\infty} |Tg_n w - Tg_n z_n| = 0\,.$$

As pointed out in Definition 4.1, this implies $\lim_{n\to\infty} d(Tg_n w, Tg_n z_n) = 0$. Since T was chosen distance preserving, we have proven

$$\lim_{n\to\infty} d(g_n w, g_n z_n) = 0\,.$$

□

Using this property one can directly show that the orbit of a compact set in $\Omega(G)$, i.e. any closed set that does not overlap $\Lambda(G)$, touches some other compact set in $\Omega(G)$ at most finitely often.

[12] i.e. an isometry on $(\hat{\mathbb{C}}, d)$. If $a, b \in \mathbb{C}$, then $d(a, b) = d(Ta, Tb)$.

Lemma 4.15. *Let G be a Kleinian group. For any two compact sets C, D in $\Omega(G)$ the set $\{g \in G \mid gC \cap D \neq \varnothing\}$ is finite.*

Proof. Assume for contradiction that $\{g \in G \mid gC \cap D \neq \varnothing\}$ is infinite. Then there exists a sequence $(g_n)_{n\in\mathbb{N}}$ of pairwise distinct elements in G and a sequence of points $(z_n)_{n\in\mathbb{N}}$ in C such that $d_n := g_n z_n \in D$. Using the compactness of C and D we can assume without loss of generality that both z_n and d_n converge. Let $w := \lim_{n\to\infty} z_n$, $d := \lim_{n\to\infty} d_n$. Note that $z, d \in (C \cup D) \subset \Omega(G)$. With triangle inequality and Lemma 4.14 one yields

$$\lim_{n\to\infty} d(g_n w, d) \leq \lim_{n\to\infty} d(g_n w, g_n z_n) + d(\overbrace{g_n z_n}^{=d_n}, d) = 0\,.$$

Hence $\lim_{n\to\infty} g_n w = d$, therefore $d \in \Lambda(G)$ which contradicts $d \in D \subset \Omega(G)$. □

For further uses we will need two theorems from [Leh64] in their original form. We will copy them here without their mathmatical derivation.

Theorem 4.16 (Theorem 2A, Ch. *Discontinuous Groups* from [Leh64])**.** *If G contains, besides the identity, only elliptic transformations, it is a finite group.*

Theorem 4.17 (Theorem 4H, Ch. *Discontinuous Groups* from [Leh64])**.** *If $\Lambda(G)$ is not a single point, it is the closure of the set of fixed points of the hyperbolic or loxodromic transformations of G.*

In particular, $\Lambda(G)$ is closed and thus is compact in $(\hat{\mathbb{C}}, d)$.

4.2 Extending the Notation for Groups as Languages

In the last section we have acquired the mathematical background knowledge that is required to derive an algorithm that generates in convergence the set $\Lambda(G)$.
We again need a some additional notation from [EPC+92] extending our word-representation for groups we have introduced in Definition 3.17.

Definition 4.18 (Cayley graph)**.** Let G be a group, and $\Sigma \subset G$ an alphabet of semigroup generators for G. The Cayley graph $\Gamma(G, \Sigma)$ is a directed, labeled graph. The set of vertices of $\Gamma(G, \Sigma)$ is G. There is an edge connecting $x \in G$ to $y \in G$ with label $\sigma \in \Sigma$ if and only if $\sigma x = y$.

Definition 4.19 (geodesic elements of a group, $|\cdot|_\Sigma$, $G_{\Sigma,n}$)**.** Let G be a group which is generated by $\Sigma \subset G$ as a semigroup.
Then we will call a word $w \in \Sigma^*$ *geodesic* if it has minimal length among all strings representing the same element as w. Such a geodesic word w can be considered as a shortest path in $\Gamma(G, \Sigma)$ from ι to $\pi(w)$. For a given $g \in G$ we will write

$$|g|_\Sigma = \min\{n \in \mathbb{N} : \exists w \in \Sigma^n \text{ such that } \pi(w) = g\}$$

for the length of its geodesic. Let $a, b \in \Sigma^*$, then clearly $|\pi(a)\pi(b)|_\Sigma = |\pi(ab)|_\Sigma \leq |\pi(a)|_\Sigma + |\pi(b)|_\Sigma$ holds. The set

$$G_{\Sigma,n} := \{\pi(w) \in G \mid w \in \Sigma^n \text{ geodesic}\} = \{g \in G \mid |g|_\Sigma = n\}$$

denotes all elements $g \in G$ which can be arrived from $\iota \in G$ at a minimum distance of precisely n steps in the Cayley graph $\Gamma(G, \Sigma)$. The sets $(G_{\Sigma,n})_{n\in\mathbb{N}}$ form a partition of G.

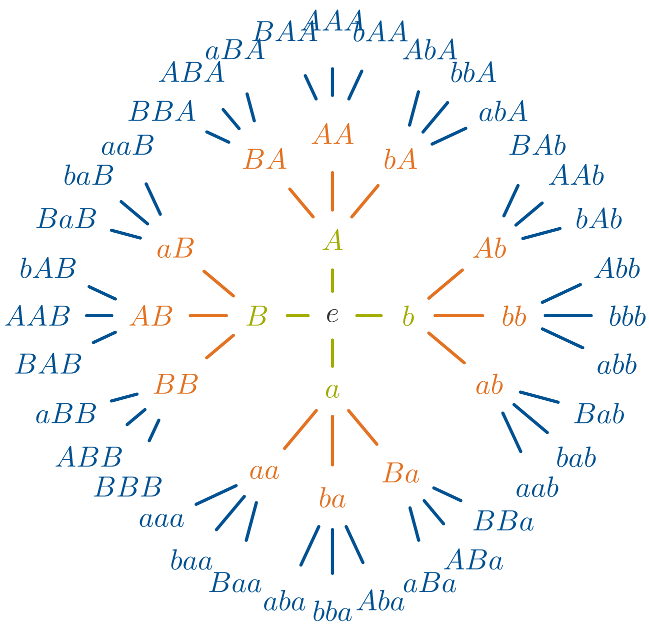


Figure 15: The Cayley graph $\Gamma(G, \Sigma)$ for the free group G with the two generators a and b. Here $\Sigma = \{a, b, A, B\}$ where $A = a^{-1}$, $B = b^{-1}$. The sets $G_{\Sigma,0}$, $G_{\Sigma,1}$, $G_{\Sigma,2}$, $G_{\Sigma,3}$ are denoted by different colors. For the free group, every word that cannot be simplified by canceling successive inverses, i.e. it does not contain the substring aA, Aa, bB or Bb, is geodesic.

4.3 Using Geodesic Group Elements for Convergence

From now on, our main goal is to prove the subsequent theorem. The idea for conjecturing this theorem came from Proposition 3.20 , which gives a similar statement for hyperbolic IFS.

Theorem 4.20 (Hausdorff convergence of $C \in \mathcal{H}(\Omega(G))$ to $\Lambda(G)$)**.** *Let G be a Kleinian group with a finite subset $\Sigma \subset G$ as semigroup generators. Furthermore, let $\Lambda(G) \neq \varnothing$ and $C \in \mathcal{H}(\Omega(G))$, or in other words, a compact set in $\hat{\mathbb{C}}$ that does not overlap $\Lambda(G)$. Then the following convergence holds on $\mathcal{H}(\hat{\mathbb{C}})$ with respect to the Hausdorff metric h:*

$$\lim_{n\to\infty} \bigcup_{g\in G_{\Sigma,n}} g(C) = \Lambda(G) \tag{12}$$

Notation 4.21. Because we do not want to use the complicated term $\bigcup_{g\in G_{\Sigma,n}} g(C)$ too often, we will write

$$G_{\Sigma,n}C := \bigcup_{g\in G_{\Sigma,n}} g(C)$$

instead.

Is the requirement $C \in \mathcal{H}(\Omega(G))$ instead of $C \in \mathcal{H}(\hat{\mathbb{C}})$ in Theorem 4.20 really necessary?

Example 4.22. Consider the Kleinian group G that is generated by the single hyperbolic transformation $h = (z \mapsto 2z)$. Let $\Sigma = \{h^{-1}, h\}$ be its set of semigroup generators. With

this $G = \{z \mapsto 2^k z \mid k \in \mathbb{Z}\}$ and $\Lambda(G) = \{0, \infty\}$. Thus $G_{\Sigma,n}C = 2^n C \cup 2^{-n} C$. Intuitively it is clear that $G_{\Sigma,n}C$ for any set $C \in \mathcal{H}(\Omega(G))$ will converge to $\Lambda(G)$ on the Riemann sphere.
What happens for $C \in \mathcal{H}(\hat{\mathbb{C}})$ in general?
If we take $C = \{0\}$, then $G_{\Sigma,n}C = \{0\}$ for all $n \in \mathbb{N}$ as every transformation in G fixes 0. Thus we cannot expect to approach every point of $\Lambda(G)$ if $C \cap \Omega(G) = \varnothing$.
Furthermore, if $C \in \mathcal{H}(\hat{\mathbb{C}})$ covers some open neighborhood of 0, then $(G_{\Sigma,n}C)_{n\in\mathbb{N}}$ inflates and will cover almost the whole Riemann sphere after a short time. In order to prevent such a behavior, it is important to require C to be bounded away from $\Lambda(G)$. Sets in $\mathcal{H}(\Omega(G))$ fulfill this requirement since $\Lambda(G)$ is closed.

In order to prove Equation (12) we will show that both $d(G_{\Sigma,n}C, \Lambda(G))$ and $d(\Lambda(G), G_{\Sigma,n}C)$ tend to 0 for $n \to \infty$.

4.3.1 Leaving the Interior Regions of the Ordinary Set

Lemma 4.23. *Let G be a Kleinian group generated by a finite $\Sigma \subset G$ and $C \in \mathcal{H}(\Omega(G))$. Then*

$$\lim_{n\to\infty} d(G_{\Sigma,n}C, \Lambda(G)) = 0\,.$$

Proof. Let $\epsilon \in \mathbb{R}_{>0}$. Now we want to show that there is an $N \in \mathbb{N}$ such that for all $n > N$ the inequality $d(G_{\Sigma,n}C, \Lambda(G)) \le \epsilon$ holds, or equivalently, the inclusion $G_{\Sigma,n}C \subset \Lambda(G) + \epsilon$ is fulfilled. Set $D := \hat{\mathbb{C}} \setminus \operatorname{int}(\Lambda(G) + \epsilon)$. The set $D \subset \Omega(G)$ is closed and therefore compact in $(\hat{\mathbb{C}}, d)$. According to Lemma 4.15 the set $\{g \in G \mid gC \cap D \neq \varnothing\}$ is finite. With this and the fact that the sets $(G_{\Sigma,n})_{n\in\mathbb{N}}$ are pairwise disjoint, the set $\{n \in \mathbb{N} \mid G_{\Sigma,n}C \cap D \neq \varnothing\}$ is finite. By choosing N as the maximum of this set, we get for all $n > N$ that $G_{\Sigma,n}C \cap D = \varnothing$ which implies

$$G_{\Sigma,n}C \subset \hat{\mathbb{C}} \setminus D \subset \Lambda(G) + \epsilon\,.$$

□

4.3.2 Approximating Every Point of the Limit Set

In order to verify the other convergence, $\lim_{n\to\infty} d(\Lambda(G), G_{\Sigma,n}C) = 0$, we need to show that $G_{\Sigma,n}C$ approaches in convergence every point of $\Lambda(G)$. Using the definition of $\Lambda(G)$, it is rather easy to see that $G_{\Sigma,n}C$ touches every tiny open neighborhood of limit points from time to time, but it is harder to establish that points of $G_{\Sigma,n}C$ remain in those neighborhoods for sufficient large n.
We overcome this problem by showing that we can get and stay arbitrarily close to the fixed points of infinite-order transformations. This will be good enough, because those fixed points are dense in $\Lambda(G)$ by Theorem 4.17.
First, we will prove that a single fixed point of a non-elliptic proper transformation is approached by the orbit of any ordinary point.

Proposition 4.24. *Let G be a Kleinian group generated by a finite $\Sigma \subset G$, $h \in G$ a parabolic or loxodromic*[13] *transformation with the attractive fixed point $Fix^+ \in \Lambda(G)$ and*

[13] or a hyperbolic transformation, which we consider as a special kind of loxodromic transformation.

$z \in \Omega(G)$. Then the n-length geodesics of G applied to z approach Fix^+ arbitrarily close for large n, or in other words,

$$\lim_{n\to\infty} d(Fix^+, G_{\Sigma,n}\{z\}) = 0$$

Proof. First we will show that $\lim_{n\to\infty} h^n g\, z = Fix^+$ for every $g \in G$.
h either is parabolic or loxodromic, we will prove $\lim_{n\to\infty} h^n g\, z = Fix^+$ for each case separately.

1st case: h is parabolic. We can write $h = T \circ (z \mapsto z + a) \circ T^{-1}$ for some $a \in \mathbb{C}_{\neq 0}$ where T is a transformation that fulfills $T(\infty) = Fix^+$. We directly see that iterating the map $(z \mapsto z + a)$ on any point approaches ∞, thus by continuity of T we get

$$\lim_{n\to\infty} h^n(g\, z) = T \lim_{n\to\infty} (z \mapsto z + a)^n (T^{-1} g\, z) = T\infty = Fix^+$$

2nd case: h is loxodromic. Let Fix^- be the repulsive fixed point of h. Then we can choose any Mobius transformation T with $T(\infty) = Fix^+$, $T(0) = Fix^-$ and have $h = T \circ (z \mapsto kz) \circ T^{-1}$ for some $k \in \mathbb{C}$, $|k| > 1$.

According to Proposition 4.8 $z \in \Omega(G)$ implies $g\, z \in \Omega(G)$. We are guaranteed that $g\, z \in \Omega(G)$ is different from Fix^-, because if $g\, z$ was some fixed point of h, then it also was a fixed point for every h^n and we could directly conclude $\lim_{n\to\infty} h^n(g\, z) = g\, z$ or $g\, z \in \Lambda(G)$ (Note that the transformations h^n are pairwise different since h is of infinite order). Hence $T^{-1} g\, z \neq 0$, which again proves

$$\lim_{n\to\infty} h^n(g\, z) = T \lim_{n\to\infty} (z \mapsto kz)^n (T^{-1} g\, z) = T\infty = Fix^+$$

Unfortunately, the convergence $\lim_{n\to\infty} h^n g\, z = Fix^+$ does not attest $\lim_{n\to\infty} d(Fix^+, G_{\Sigma,n}\{z\}) = 0$, because we cannot assume that every $G_{\Sigma,n}$ contains some power of h. We have to construct a sequence of transformations with the nth transformation being in $G_{\Sigma,n}$ for which this convergence property holds.
We can represent $h = \pi(\sigma_1 \dots \sigma_k)$ with $\sigma_1, \dots, \sigma_k \in \Sigma$ by a string over the alphabet Σ. Let $w(t)$ denote the t-letter string which is obtained by taking the first t letters of the infinite repeating string $\sigma_1 \dots \sigma_k \sigma_1 \dots \sigma_k \dots$, or in mathematical language we set

$$\begin{aligned} w &: \mathbb{N} \to \Sigma^* \\ &t \mapsto w(t) \in \Sigma^t \qquad \text{where } w(t)_i = \sigma_{(i-1 \bmod k)+1} \text{ for } i \in [t] \;. \end{aligned}$$

For instance, it turns out that $\pi(w(k)) = h$, or more generally, any $t = k \cdot n + l \in \mathbb{N}$ with $l \in [k-1]$ gives us

$$\pi(w(k \cdot n + l)) = h^n \pi(\sigma_1) \dots \pi(\sigma_l)\,.$$

The order of h is infinite because it is parabolic or loxodromic (To see this, consider the conjugate transformation as above). As a consequence for a fixed $l \in [k-1]$ the sequence $(\pi(w(k \cdot n + l)))_{n\in\mathbb{N}} = (h^n \pi(\sigma_1) \dots \pi(\sigma_l))_{n\in\mathbb{N}}$ consists of pairwise different elements, which implies $\lim_{n\to\infty} |\pi(w(k \cdot n + l))|_\Sigma = \infty$ (Again, we use the fact that there is only a finite number of strings of bounded length). Furthermore $(|\pi(w(t))|_\Sigma)_{t\in\mathbb{N}}$ does not skip any natural number because

$$|\pi(w(t+1))|_\Sigma = |\pi(w(t))\pi(\sigma_{(t \bmod k)+1})|_\Sigma \leq |\pi(w(t))|_\Sigma + 1$$

So we have proven that $\lim_{t\to\infty} |\pi(w(t))|_\Sigma = \infty$ and, even more, every natural number occurs in the sequence $|\pi(w(t))|_\Sigma$. So we can choose a sequence $(t_n)_{n\in\mathbb{N}}$ for which $|\pi(w(t_n))|_\Sigma = n$ holds and take $\lim_{n\to\infty} t_n = \infty$ for granted.
For a fixed $l \in [k-1]$, we conclude by the convergence property discussed above:

$$\lim_{n\to\infty} \pi(w(k\cdot n + l))z = \lim_{n\to\infty} h^n(\pi(\sigma_1)\dots\pi(\sigma_l)z) = Fix^+ ,$$

thus the merged sequence fulfills $\lim_{t\to\infty} \pi(w(t))z = Fix^+$, which immediately implies $\lim_{n\to\infty} \pi(w(t_n))z = Fix^+$. As $\pi(w(t_n)) \in G_{\Sigma,n}$ we have finally verified that

$$\lim_{n\to\infty} d(Fix^+, G_{\Sigma,n}\{z\}) = \lim_{n\to\infty} \inf\{d(Fix^+, gz) \mid g \in G_{\Sigma,n}\} = 0$$

□

Now we are ready to prove the missing part for our eagerly awaited Theorem 4.20.

Lemma 4.25. *Let G be a Kleinian group, with $\Lambda(G) \neq \varnothing$, $C \in \mathcal{H}(\Omega(G))$. Then*

$$\lim_{n\to\infty} d(\Lambda(G), G_{\Sigma,n}C) = 0$$

Proof. Theorem 4.16 guarantees us that there are some parabolic or loxodromic transformations in G. Otherwise G was finite, which caused $\Lambda(G)$ to become the empty set.
If G contains some loxodromic transformation, then both of its fixed points are in $\Lambda(G)$, thus $|\Lambda(G)| \geq 2$. We will handle this case later. So the case $|\Lambda(G)| = 1$ occurs only if there exists a parabolic transformation in G with a fixed point Fix^+. Since this parabolic transformation is of infinite order it turns out that $\{Fix^+\} = \Lambda(G)$. By Proposition 4.24 this point is approached in convergence by some $z \in C$, thus

$$\lim_{n\to\infty} d(\Lambda(G), G_{\Sigma,n}C) \leq \lim_{n\to\infty} d(\{Fix^+\}, G_{\Sigma,n}\{z\}) = 0$$

and we are done.
Now, we are going to prove the harder and more general case $|\Lambda(G)| \geq 2$.
Let $\epsilon \in \mathbb{R}_{>0}$. Our job is to establish the existence of an $N \in \mathbb{N}$ such that for all $n > N$

$$d(\Lambda(G), G_{\Sigma,n}C) \leq \epsilon \quad \text{or equivalently,} \quad \Lambda(G) \subset G_{\Sigma,n}C + \epsilon \tag{13}$$

holds.
Assuming $|\Lambda(G)| \geq 2$ we can apply Theorem 4.17, stating that the fixed points of loxodromic transformations in G are dense in $\Lambda(G)$. Without loss of generality, we might use the attractive fixed points only, because a repulsive fixed point of some loxodromic transformation $h \in G$ becomes an attractive fixed point of $h^{-1} \in G$ (see [MSW02]). The density of those attractive fixed points enables us to cover $\Lambda(G)$ with open $\frac{\epsilon}{2}$-balls with the attractive fixed points as their center.
By a well known result of Topology, every open cover of a compact set has a finite subcover. (for a proof see for example [Bro13]). Thus the compact set $\Lambda(G)$ (for compactness compare Theorem 4.17) therefore can be covered as follows

$$\Lambda(G) \subset \bigcup_{k=1}^{K} B(Fix_k^+, \frac{\epsilon}{2}) , \tag{14}$$

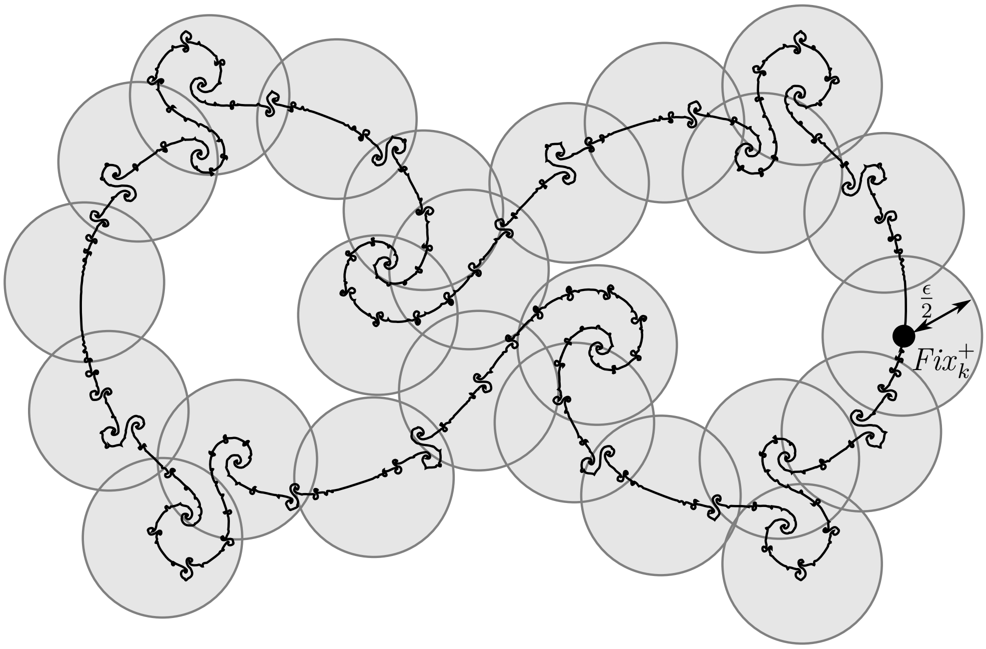


Figure 16: Sketch of the succeeding proof. There is a finite cover of $\Lambda(G)$ with open $\frac{\epsilon}{2}$-balls having attractive fixed points of loxodromic transformations as their center. Then there is an $N \in \mathbb{N}$ such that for every $n > N$ there is at least one element of $G_{\Sigma,n}\{z\}$ within every ball.

where $K \in \mathbb{N}$ and $h_1, \ldots h_K \in G$ is some finite collection of loxodromic transformations with the attractive fixed points $Fix_1^+, \ldots, Fix_K^+$.
Let $z \in C \subset \Omega(G)$ be some ordinary point in C. Then Proposition 4.24 ensures that $G_{\Sigma,n}\{z\}$ for big n will approach every tiny open neighborhood of the fixed point of a loxodromic transformation in G and $G_{\Sigma,n}\{z\}$ will remain within any of those neighborhoods. So for the loxodromics h_k with $k \in \{1, \ldots, K\}$ there exists N_k such that for all $n > N_k$

$$d(Fix_k^+, G_{\Sigma,n}\{z\}) \leq \frac{\epsilon}{2} \quad \text{or equivalently,} \quad Fix_k^+ \in G_{\Sigma,n}\{z\} + \frac{\epsilon}{2} \tag{15}$$

holds. Now we are done by choosing $N := \max\{N_1, \ldots, N_K\}$, because this gives for all $n > N$ the inclusion in Equation (13): Let $x \in \Lambda(G)$, then by Section 4.3.2 there is a $k \in \{1, \ldots, K\}$ such that $d(x, Fix_k^+) \leq \frac{\epsilon}{2}$ and by Equation (15) there is a $y \in G_{\Sigma,n}\{z\}$ with $d(Fix_k^+, y) \leq \frac{\epsilon}{2}$. Triangle inequality gives us $d(x, y) \leq d(x, Fix_k^+) + d(Fix_k^+, y) \leq \epsilon$, therefore

$$x \in G_{\Sigma,n}\{z\} + \epsilon \subset G_{\Sigma,n}C + \epsilon\,.$$

□

This was the missing link in order to prove Theorem 4.20

Proof of Theorem 4.20. $\Lambda(G)$ is non-empty by assumption and it is closed by Theorem 4.17. Thus $\Lambda(G) \in \mathcal{H}(\hat{\mathbb{C}})$. On the one hand, under the given assumptions, we were able to show (see Lemma 4.23) $\lim_{n\to\infty} d(G_{\Sigma,n}C, \Lambda(G)) = 0$ and on the other hand (see Lemma 4.25)

$\lim_{n\to\infty} d(\Lambda(G), G_{\Sigma,n}C) = 0$. If we plug those properties together, then we finally get

$$\begin{aligned} h(G_{\Sigma,n}C, \Lambda(G)) &= \lim_{n\to\infty} \max\{d(G_{\Sigma,n}C, \Lambda(G)), d(\Lambda(G), G_{\Sigma,n}C)\} \\ &= \max\{0,0\} = 0\,. \end{aligned}$$

Thus $\lim_{n\to\infty} G_{\Sigma,n}C = \Lambda(G)$ in $(\mathcal{H}(\hat{\mathbb{C}}), h)$. □

4.4 Utilizing Automatons Accepting the Language of Geodesics

How can we calculate the sets $G_{\Sigma,n}C$ progressively, which are needed in Theorem 4.20? For a hyperbolic IFS the analogous was very simple: Here, we were able to compute $W^{n+1}C$ directly by applying the Hutchinson operator W to the set W^nC.
With a more sophisticated method, the progressive computation of the sets $G_{\Sigma,n}C$ is possible, provided that the language of geodesics of the groups is accepted by a deterministic finite automaton.

Definition 4.26 (adapted from Ch. 2. in [Hop07]: deterministic finite automaton, language of a DFA, regular languages, reversal of words and languages)**.** A five-tuple $A = (Q, \Sigma, \delta, q_0, F)$ is called a *deterministic finite automaton* (DFA) where Q is a finite set of *states*, Σ a finite alphabet of *input symbols*, $\delta : Q \times \Sigma \to Q$ a *transition function*, $q_0 \in Q$ its *start state* and $F \subset Q$ its set of *accepting* states. The *extended transition function* $\hat{\delta} : Q \times \Sigma^* \to Q$ tells us the state where we land if we successively apply for each letter of the input string the transition function δ to get from one state to another. It is inductively defined by $\hat{\delta}(q, \epsilon) = q$ and $\hat{\delta}(q, wa) = \delta(\hat{\delta}(q, w), a)$ where $q \in Q$, $w \in \Sigma^*$, $a \in \Sigma$. The DFA A *accepts* the language

$$L(A) := \{w \in \Sigma^* \mid \hat{\delta}(q_0, w) \in F\}\,.$$

A language $L \subset \Sigma^*$ is called *regular*, if there exists a DFA $A = (Q, \Sigma, \delta, q_0, F)$ such that $L = L(A)$.
The *reversal* w^R of a string $w = a_1 a_2 \dots a_n \in \Sigma$ is it written backwards, i.e. $w^R = a_n a_{n-1} \dots a_1$. The reversal $L^R := \{w^R \in \Sigma^* \mid w \in L\}$ of a language $L \subset \Sigma^*$ consists of all its reversed strings.

For our method, we will need the following theorem, which we will not prove here.

Theorem 4.27 (Theorem 4.11 from [Hop07])**.** *If L is a regular language, so is L^R.*

Our algorithm will be based on the fact that for almost every Kleinian group there is an automaton which accepts the language of the geodesics in G:

Theorem 4.28 (Theorem 3.4.5 from [EPC+92])**.** *Let G be a word hyperbolic group and let Σ be a set of semigroup generators closed under inversion. The geodesics over Σ form a regular language, which is part of an automatic structure.*

We will not go into the details here to explain the meaning of a hyperbolic group and under what circumstances a Kleinian group is hyperbolic as it would go beyond the scope of this thesis. A good reference is [BH99]. For free groups we can use the automaton depicted in Figure 17.

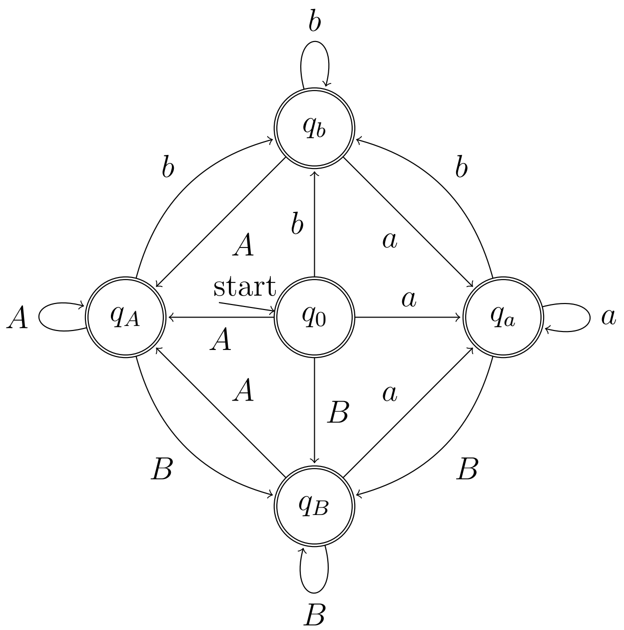


Figure 17: A graph representing a 5-state automaton that accepts the reversed geodesics for a two generator free group. The language of reversed geodesics consists of every word over $\{a, A, b, B\}$ that does not contain the substrings aA, Aa, bB and Bb. Each arrow stands for a transition. Every state is an accepting state.

In the following, let G be a Kleinian group, Σ a set of semigroup generators and $A = (Q, \Sigma, \delta, q_0, F)$ a DFA that accepts the language of all reversed geodesics of G, i.e.

$$L(A) = \{w \in \Sigma^* \mid w \text{ geodesic in } G\}^R .$$

Then partitioning Σ^* yields to a precious relation. We define for every state $q \in Q$ and every length $n \in \mathbb{N}$

$$L_{q,n} := \{w \in \Sigma^n \mid \hat{\delta}(q_0, w^R) = q\} .$$

With this, we obviously have

$$\bigcup_{q \in F} L_{q,n} = L(A)^R \cap \Sigma^n = \{w \in \Sigma^n \mid w \text{ geodesic in } G\}^R . \tag{16}$$

Furthermore, we can calculate the languages $(L_{q,n})_{q \in Q, n \in \mathbb{N}}$ recursively:

$$L_{q,0} = \{w \in \Sigma^0 \mid \hat{\delta}(q_0, w^R) = q\} = \begin{cases} \{\epsilon\} & \text{if } q = q_0 \\ \varnothing & \text{if } q \neq q_0 \end{cases} \tag{17}$$

$$\begin{aligned} L_{q,n+1} &= \{aw \in \Sigma^{n+1} \mid a \in \Sigma,\ w \in \Sigma^n : \hat{\delta}(q_0, \underbrace{(aw)^R}_{=w^R a}) = q\} \\ &= \{aw \in \Sigma^{n+1} \mid a \in \Sigma,\ w \in \Sigma^n,\ p \in Q : \hat{\delta}(q_0, w^R) = p \wedge \delta(p, a) = q\} \\ &= \bigcup_{\substack{a \in \Sigma,\, p \in Q: \\ \delta(p,a) = q}} a\, L_{p,n} \end{aligned} \tag{18}$$

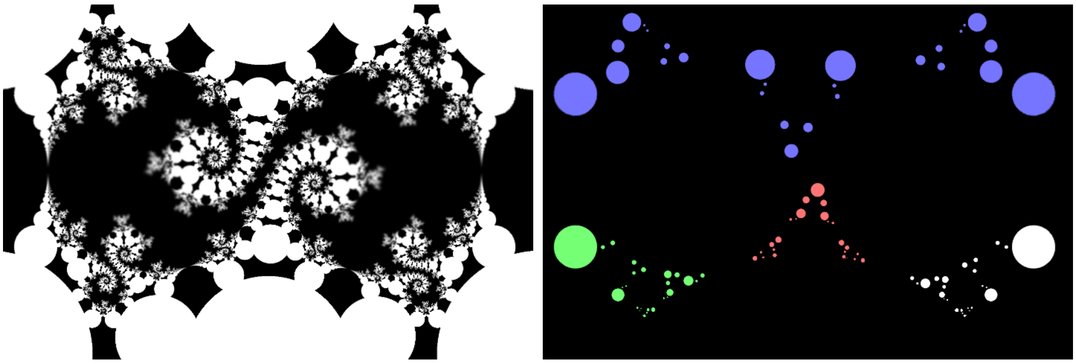


Figure 18: Screenshots of the implementation. The right pictures shows the sets $(W_{q,4}C)_{q\in Q}$ each state in a different color.

Now for $q \in Q, n \in \mathbb{N}$ and $C \in \mathcal{H}(\hat{\mathbb{C}})$ we set,

$$W_{q,n}C := \bigcup_{w\in L_{q,n}} \pi(w)C\,.$$

By Equation (16) we immediately see that

$$\bigcup_{q\in F} W_{q,n}C = G_{\Sigma,n}C$$

and, furthermore, by Equations (17) and (18)

$$W_{q,0}C = \begin{cases} C & \text{if } q = q_0 \\ \varnothing & \text{if } q \neq q_0 \end{cases} \qquad W_{q,n+1}C = \bigcup_{\substack{a\in\Sigma,\, p\in Q:\\ \delta(p,a)=q}} \pi(a\,L_{p,n})C = \bigcup_{\substack{a\in\Sigma,\, p\in Q:\\ \delta(p,a)=q}} \pi(a)W_{p,n}C\,.$$

So in combination with Theorem 4.20 we have proven:

Theorem 4.29. *Let G be a Kleinian group with $\Lambda(G) \neq \varnothing$, $C \in \mathcal{H}(\Omega(G))$ and let Σ be a set of semigroup generators that is closed under inversion. Furthermore, suppose that there is a DFA $A = (Q, \Sigma, \delta, q_0, F)$ with $L(A) = \{w \in \Sigma^* \mid w \text{ geodesic in } G\}^R$.*
Then the sequence $(W_{q,n}C)_{q\in Q, n\in\mathbb{N}} \subset H(\hat{\mathbb{C}})$ which is recursively defined by

$$W_{q,0}C = \begin{cases} C & \textit{if } q = q_0 \\ \varnothing & \textit{if } q \neq q_0 \end{cases} \qquad W_{q,n+1}C = \bigcup_{\substack{a\in\Sigma,\, p\in Q:\\ \delta(p,a)=q}} \pi(a)W_{p,n}C \tag{19}$$

fulfills

$$\lim_{n\to\infty} \bigcup_{q\in F} W_{q,n}C = \lim_{n\to\infty} G_{\Sigma,n}C = \Lambda(G)\,.$$

4.4.1 Textured Based Implementation of Automaton based Approach

Suppose we are in the setting of Theorem 4.29. Now let us pour Equation (19) into an algorithm, which then gives us a sequence of sets whose union converges to the limit set.

We will represent a set $C \in \mathcal{H}(\hat{\mathbb{C}})$ by a high resolution single-channel texture. The texture captures a sufficient large rectangular sector of $\mathbb{C} \subset \hat{\mathbb{C}}$. Pixels that represent points in C will assigned to the color 1 (white) and all other pixels will assigned to 0 (black).

Algorithm 3: A method to calculate and visualize $G_{\Sigma,n}C$ progressively

Initialize two high resolution $|Q|$-channel textures $(\mathsf{CurrentW}_q)_{q\in Q}$ and $(\mathsf{PreviousW}_q)_{q\in Q}$ of the same size. (Or, alternatively, initialize $2 \cdot |Q|$ single channel textures)
foreach *pixel* p_x*, state* $q \in Q$ **do**
  $x \leftarrow$ complex number in $\mathbb{C}$ that corresponds to the pixel p_x
  $\mathsf{CurrentW}_q(p_x) \leftarrow 1$ if $q = q_0$ and $x \in C$, otherwise $\mathsf{CurrentW}_q(p_x) \leftarrow 0$
end
while *program is running* **do at most (roughly)** 30 **times a second**
  $\mathsf{PreviousW} \leftarrow \mathsf{CurrentW}$
  `/* The rendering procedure for a single frame */`
  `/* For every` $q \in Q$ `set` $W_{q,n+1}C$ `to` $\bigcup_{\substack{a\in\Sigma,\, p\in Q:\\ \delta(p,a)=q}} \pi(a)W_{p,n}C$ `*/`
  foreach *pixel* p_x **do**
    $\mathsf{CurrentW}_q(p_x) = 0$ for every state $q \in Q$
    $x \leftarrow$ complex number in $\mathbb{C}$ that corresponds to the pixel p_x
    foreach *state* $p, q \in Q$*, letter* $a \in \Sigma$ *with* $\delta(p, a) = q$ **do**
      $y \leftarrow \pi(a)^{-1}x$ `/* Apply the corresponding Möbius transformation */`
      $p_y \leftarrow$ pixel(coordinate) that corresponds to the point y
      $\mathsf{CurrentW}_q(p_x) \leftarrow \max(\mathsf{CurrentW}_q(p_x), \mathsf{PreviousW}_p(p_y))$ `/* Take the union. */`
    end
  end
  Display the overlay of the textures $(\mathsf{CurrentW}_q)_{q\in F}$ `/* Display` $\bigcup_{q\in F} W_{q,n+1}C$ `*/`
end

We have implemented an example for a free two generator Kleinian group G here: `http://aaron.montag.info/ba/14`using the automaton from Figure 17. The two generators were chosen using *grandma's receipt* from [MSW02] with the parameters $t_a = 1.91 + 0.05i$, $t_b = 3$.

In this case, the limit set $\Lambda(G)$ is bounded by 1. Therefore we were able to choose $C := \{z \in \hat{\mathbb{C}} \mid |z| \geq 1.1\} \subset \Omega(G)$, which is a compact set of $\hat{\mathbb{C}}$. In the example we have decided to keep visually track of the all the sets that once occurred in $G_n\Sigma$ for some $n \in \mathbb{N}$ because the set $G_n\Sigma$ would vanish too fast on the screen for big n. The user might disable keeping track by regulating the illumination-slider. The states of the automaton accepting the geodesic language of a free group were encoded by the different color channels of a single texture[14]. With the color-slider these states can be made visible.

[14]Actually there are the *four* channels red, green, blue and alpha for an automaton with *five* states. We kept out the state q_0 and poured the initial lightness for C directly in the color channels for q_a, q_A, q_b, q_B which makes no difference for the visualized image.

4.5 Measure-Based Approach to Render Kleinian Groups

We will need a statement about the Lebesgue measure of the limit set here.

Theorem 4.30 (Ahlfors Conjecture, Theorem 5.6.6 in [Mar07]). *For any finitely generated group G, either $\Lambda(G) = \hat{\mathbb{C}}$, or $\Lambda(G)$ has 2-dimensional Lebesgue measure zero.*

According to [Mar07] the missing link for Ahlfors conjecture to hold was finally given in 2004 by Ian Agol with his proof of the Tameness theorem. Since we required $\Omega(G) \neq \varnothing$ in our definition of Kleinian groups, the conjecture implies $\mathcal{L}(\Lambda(G)) = 0$ for every finitely generated Kleinian group in our sense.
The last algorithm had two big drawbacks. A major disadvantage is that after some time the displayed set appears to vanish. This is a consequence of the nature of the limit set itself, which has according to Ahlfors conjecture Lebesgue measure 0.
Another big drawback lies in the fact that we had to choose some set $C \in \mathcal{H}(\Omega(G))$ instead of any compact set of $\mathcal{H}(\hat{\mathbb{C}})$. On the one hand, some a priori knowledge of the limit set $\Lambda(G)$ is required to choose such a set. On the other hand the user cannot simply change parameters of the group at running time, because by doing so he might move $\Lambda(G)$ over the current displayed set and "messing up" everything.
Similar as in Section 3.3, we can use measures instead of sets. Likewise these measures will be encoded as textures. It turns out that this gives the opportunity to drop the disturbing property of choosing an initial set C that does not overlap $\Lambda(G)$.
One option to do so is simulating the random IFS algorithm: Let G be a Kleinian group finitely generated by $\Sigma \subset G$ as semigroup, where each $\sigma \in \Sigma$ is assigned a probability $p_\sigma \in (0, 1]$ such that $\sum_{\sigma\in\Sigma} p_\sigma = 1$. The random IFS algorithm outputs the trace of a single point that is iteratively transformed by some random element $\sigma \in \Sigma$ which is chosen for each iteration independently with probability p_σ. Similar as in Definition 3.15 this process can be simulated by iterating the Markov operator that takes some probability measure $\mu \in \mathcal{M}^1(\hat{\mathbb{C}})$ and maps it to the new measure $\sum_{\sigma\in\Sigma}^N p_\sigma \cdot \sigma\mu$. The results of some test implementation [15] were not satisfying as in several Kleinian groups there are some regions of the limit sets that have some regions that are very unlikely to be visited by the random IFS algorithm and therefore are assigned an almost invisible small probability.
Using measures we can attain much better images of the limit set if we use our results developed for the set based approach (see algorithm 3) by simply replacing sets with measures and the union of sets with the sum of measures. Again, we assume that Σ is a finite set of semigroup generators for the Kleinian group G and $A = (Q, \Sigma, \delta, q_0, F)$ is a DFA accepting the language of all reversed geodesics of G. We define the languages $(L_{q,n})_{q\in Q, n\in\mathbb{N}}$ as in Section 4.4. Now for a measure $\mu \in M(\hat{\mathbb{C}})$ we set

$$M_{q,n}\mu := \sum_{w\in L_{q,n}} \pi(w)\mu\,.$$

Analogously as in Theorem 4.29, $M_{q,n}\mu$ can be computed recursively by

$$M_{q,0}\mu = \begin{cases} \mu & \text{if } q = q_0 \\ 0 & \text{if } q \neq q_0 \end{cases} \qquad\qquad M_{q,n+1}\mu = \sum_{\substack{a\in\Sigma,\, p\in Q:\\ \delta(p,a)=q}} \pi(a) M_{p,n}\mu\,.$$

[15] see `http://aaron.montag.info/ba/15`.

and it fulfills

$$\sum_{q\in F} M_{q,n}\mu = \sum_{g\in G_{\Sigma,n}} g\mu =: G_{\Sigma,n}\mu\,.$$

How does $G_{\Sigma,n}\mu$ look like for big $n \in \mathbb{N}$ if μ is an arbitrary measure? We cannot expect $G_{\Sigma,n}\mu$ to convergence in some proper sense since $G_{\Sigma,n}\mu(\hat{\mathbb{C}}) = \mu(\hat{\mathbb{C}}) \cdot |G_{\Sigma,n}|$, which tends to infinity for $\mu(\hat{\mathbb{C}}) \neq 0$ and growing sets $G_{\Sigma,n}$. Can we expect that its support will approach $\Lambda(G)$? Not in general: A counterexample for the group that is generated by a single hyperbolic Möbius transformation can be given by choosing the Dirac measure $\mu = \delta_z$ where $z \in \Lambda(G)$ is one of the two fixed points of the Möbius transformation. Then the support of the measure $G_{\Sigma,n}\delta_z = 2^n\delta_z$ does not attain the other fixed point.
Under suitable assumptions on the measure a positive answer is given by the following theorem:

Theorem 4.31. *Let $\mu \in M(\hat{\mathbb{C}})$ be a measure that is bounded by a multiple of the Lebesgue measure, i.e. there is a constant $c \in \mathbb{R}_{>0}$ such that for every Borel set B*

$$\mu(B) \leq c \cdot \mathcal{L}(B)$$

holds.
Then we have for every compact set $C \in \mathcal{H}(\Omega(G))$

$$\lim_{n\to\infty} G_{\Sigma,n}\mu(C) = 0$$

and for every open set $O \subset \mathbb{C}$ with $O \cap \Lambda(G) \neq \varnothing$

$$\liminf_{n\to\infty} G_{\Sigma,n}\mu(O) \geq \frac{\mu(\hat{\mathbb{C}})}{2}\,.$$

Proof. First let us prove $\lim_{n\to\infty} G_{\Sigma,n}\mu(C) = 0$. Using the definition of transformed measures and characteristic functions one may write

$$G_{\Sigma,n}\mu(C) = \sum_{g\in G_{\Sigma,n}} \mu(g^{-1}C) = \int_{\hat{\mathbb{C}}} \sum_{g\in G_{\Sigma,n}} \chi_{g^{-1}C}(z)\, d\mu(z) \tag{20}$$

The set $\{g \in G \mid gC \cap C \neq \varnothing\}$ has a finite cardinality K according to Lemma 4.15 . Since Möbius transformations are bijective we may conclude that for every $g^{-1} \in G_{\Sigma,n}$ the set $\{h \in G_{\Sigma,n} \mid h^{-1}C \cap g^{-1}C \neq \varnothing\}$ has a cardinality less or equal to K. Hence the sum $\sum_{g\in G_{\Sigma,n}} \chi_{g^{-1}C}(z)$ can be bounded above by K. Furthermore, this sum has the support $G_{\Sigma^{-1},n}C$. With this we can bound Equation (20) from above to

$$G_{\Sigma,n}\mu(C) \leq K\mu(G_{\Sigma^{-1},n}C) \leq Kc\mathcal{L}(G_{\Sigma^{-1},n}C) \tag{21}$$

By Lemma 4.23 $\lim_{n\to\infty} d(G_{\Sigma^{-1},n}C, \Lambda(G)) = 0$ which implies that there is a zero-sequence $(\epsilon_n)_{n\in\mathbb{N}}$ such that $G_{\Sigma^{-1},n}C \subset \Lambda(G) + \epsilon_n$. Since $\Lambda(G)$ is closed, $\Lambda(G) = \bigcap_{n=1}^{\infty}(\Lambda(G) + \epsilon_n)$ holds. Measures are continuous from above, therefore $\lim_{n\to\infty} \mathcal{L}(\Lambda(G) + \epsilon_n) = \mathcal{L}(\Lambda(G))$. By taking the limit of Equation (21) and Ahlfors conjecture we finally yield:

$$\lim_{n\to\infty} G_{\Sigma,n}\mu(C) \leq Kc \lim_{n\to\infty} \mathcal{L}(G_{\Sigma^{-1},n}C) \leq Kc \lim_{n\to\infty} \mathcal{L}(\Lambda(G) + \epsilon_n) = Kc\mathcal{L}(\Lambda(G)) = 0\,,$$

hence $\lim_{n\to\infty} G_{\Sigma,n}\mu(C) = 0$.

Now let $O \subset \hat{\mathbb{C}}$ be an open set that has a non-empty intersection with $\Lambda(G)$. In this setting $\Lambda(G) \neq \varnothing$. As discussed in the proof of Lemma 4.25 $\Lambda(G)$ is either a single point of a parabolic transformation or it is the closure of the fixed points of loxodromic transformations. In any case, there is a parabolic or loxodromic transformation h having a attractive fixed point $Fix^+ \in O$. Since O is open, there is a neighborhood $B(Fix^+, \epsilon) \subset O$.

With little afford Proposition 4.24 can be generalized to the property that there exists is a sequence of transformations $(g_n \in G_{\Sigma,n})_{n\in\mathbb{N}}$ which uniformly attracts any compact set in $\Omega(G)$ to Fix^+. We will not give an explicit proof here as it is almost a technical adaption of Proposition 4.24.

As shown above we have the convergence

$$\lim_{\epsilon\to 0} \mu(\Lambda(G) + \epsilon) \leq \lim_{\epsilon\to 0} c\mathcal{L}(\Lambda(G) + \epsilon) = 0$$

So, there is a $\delta > 0$ such that $\mu(\Lambda(G) + \delta) \leq \frac{\mu(\hat{\mathbb{C}})}{2}$. The compact set $C := \hat{\mathbb{C}} \setminus \mathrm{int}(\Lambda(G) + \delta)$ fulfills the inequality $\mu(C) \geq \frac{\mu(\hat{\mathbb{C}})}{2}$.

By the generalized convergence property, there is an N such that $g_n C \subset B(Fix^+, \epsilon)$ for every $n > N$. For those n we can generously estimate from below

$$G_{\Sigma,n}\mu(O) \geq G_{\Sigma,n}\mu(B(Fix^+, \epsilon)) \geq \mu(g_n^{-1} B(Fix^+, \epsilon)) \geq \mu(C) \geq \frac{\mu(\hat{\mathbb{C}})}{2}.$$

This finished our proof.

□

Theorem 4.31 let us expect good results if we calculate $G_{\Sigma,n}\mu$ where μ is some uniform distribution on the region covered by the screen: Then μ vanishes exactly at those regions that are not covered by $\Lambda(G)$.

4.5.1 Implementation

The computation of $G_{\Sigma,n}\mu$ for a measure μ is analogous to the computation of $G_{\Sigma,n}C$ for a set C in algorithm 3. The densities of the occurring measures $M_{q,n}\mu = \sum_{w\in L_{q,n}} \pi(w)\mu$ will be encoded in textures as described in Section 3.3.1.

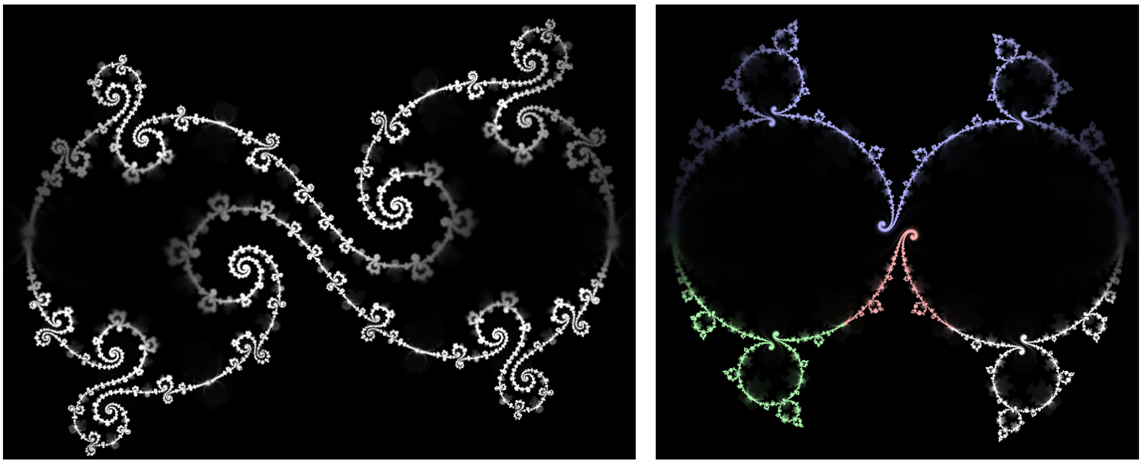


Figure 19: Screenshots of the measure-based implementation. The right picture again visualized the internal states.

With this we yield the following algorithm:

Algorithm 4: A method to calculate and visualize $G_{\Sigma,n}\mu$ progressively

Initialize two high resolution $|Q|$-channel textures $(\mathsf{CurrentD}_q)_{q\in Q}$ and $(\mathsf{PreviousD}_q)_{q\in Q}$ of the same size. (Or, alternatively, initialize $2\cdot|Q|$ single channel textures)

foreach *pixel* p_x, *state* $q \in Q$ **do**
    $\mathsf{CurrentD}_q(p_x) \leftarrow 1$ if $q = q_0$, otherwise $\mathsf{CurrentD}_q(p_x) \leftarrow 0$
end
while *program is running* **do at most (roughly)** 30 **times a second**
    $\mathsf{PreviousD} \leftarrow \mathsf{CurrentD}$
    `/* The rendering procedure for a single frame */`
    `/* For every` $q \in Q$ `set` $M_{q,n+1}\mu$ `to` $\sum_{\substack{a\in\Sigma,\, p\in Q:\\ \delta(p,a)=q}} \pi(a)M_{p,n}\mu$ `*/`
    foreach *pixel* p_x **do**
        $\mathsf{CurrentD}_q(p_x) = 0$ for every state $q \in Q$
        $x \leftarrow$ complex number in $\mathbb{C}$ that corresponds to the pixel p_x
        foreach *state* $p, q \in Q$, *letter* $a \in \Sigma$ *with* $\delta(p,a) = q$ **do**
            $y \leftarrow \pi(a)^{-1}x$ `/* Apply the corresponding Möbius transformation */`
            $p_y \leftarrow$ pixel(coordinate) that corresponds to the point y
            $\mathsf{CurrentD}_q(p_x) \leftarrow \mathsf{CurrentD}_q(p_x) + |\det D\pi(a)^{-1}(x)| \cdot \mathsf{PreviousD}_p(p_y)$
        end
    end
    Display the overlay of the textures $(\mathsf{CurrentD}_q)_{q\in F}$ `/* Display` $\sum_{q\in F} M_{q,n+1}\mu$ `*/`
end

An example implementation can be found at: `http://aaron.montag.info/ba/16`. Here the user can change parameters at running time.

4.6 Conclusion

The results are better than the images generated by algorithm 3. Nevertheless there are still some blurred regions caused by the magnifications of the texture with limited resolution. Furthermore it is necessary to cover the whole limit set by the screen.
But if one is willing to accept these drawbacks, this rendering method provides an environment for visualizing the limit sets effectively. In particular, the effects of parameter change become visible instantaneously.

References

[Bar12] Michael F Barnsley. *Fractals everywhere.* Courier Dover Publications, 2012.

[BH99] Martin R Bridson and André Haefliger. *Metric spaces of non-positive curvature*, volume 319. Springer, 1999.

[Bro13] Martin Brokate. Funktionalanalysis, WS 2013/14. lecture notes, available at `http://www-m6.ma.tum.de/~brokate/`; accessed 2-April-2014, 2013.

[Els04] Jürgen Elstrodt. *Maß-und Integrationstheorie.* Springer DE, 2004.

[EPC+92] David B. A. Epstein, M. S. Paterson, J. W. Cannon, D. F. Holt, S. V. Levy, and W. P. Thurston. *Word Processing in Groups.* A. K. Peters, Ltd., Natick, MA, USA, 1992.

[HH99] Katharina Habermann and Lutz Habermann. Einführung in die Theorie der Kleinschen Gruppen. available at `http://service.ifam.uni-hannover.de/~habermann/skripte/einfklein.pdf`; accessed 2-April-2014, 1999.

[Hop07] John E Hopcroft. *Introduction to automata theory, languages, and computation.* Pearson Addison Wesley, 2007.

[HPS91] Daryl Hepting, Przemyslaw Prusinkiewicz, and Dietmar Saupe. Rendering methods for iterated function systems. *Fractals in the fundamental and applied sciences*, pages 183–224, 1991.

[Hut79] John E Hutchinson. *Fractals and self similarity.* University of Melbourne., 1979.

[Kön03] Konrad Königsberger. *Analysis 1*, volume 1. Springer DE, 2003.

[Leh64] Joseph Lehner. *Discontinuous groups and automorphic functions*, volume 8. American Mathematical Society Providence, 1964.

[Mar07] Albert Marden. *Outer circles: An introduction to hyperbolic 3-manifolds.* Cambridge University Press, 2007.

[Mat13] Daniel Matthes. Vorlesungsvorbereitung Gewohnliche Differentialgleichungen, SS 2013. lecture notes, available at `http://www-lm.ma.tum.de/gdgl13/folien.html`; accessed 12-April-2014, 2013.

[MSW02] David Mumford, Caroline Series, and David Wright. *Indra's pearls: the vision of Felix Klein.* Cambridge University Press, 2002.

[Nee11] Tristan Needham. *Anschauliche Funktionentheorie.* Oldenbourg Verlag, 2011.

[RG11] Jürgen Richter-Gebert. *Perspectives on projective geometry: a guided tour through real and complex geometry.* Springer, 2011.